FAILY, S., FLÉCHAIS, I. and COLES-KEMP, L. (eds.) 2012. *Proceedings of the Designing interactive secure systems workshop (DISS 2012), part of the 26th International BCS human computer interaction conference (HCI 2012): people and computers, 11 September 2012, Birmingham, UK*. Swindon: BCS [online], article numbers 62-70. Preface available from: https://doi.org/10.14236/ewic/HCI2012.70

Proceedings of the Designing interactive secure systems workshop (DISS 2012).

FAILY, S., FLÉCHAIS, I. and COLES-KEMP, L. (eds.)

2012

Full bibliographic details of individual papers are available on the ScienceOpen database. DOI links to the webpages for individual papers are provided on the table of contents.

Table of Contents

Designing Interactive Secure Systems: Workshop at British HCI 2012

Shamal Faily
Department of Computer Science
University of Oxford
shamal.faily@cs.ox.ac.uk

Ivan Fléchais
Department of Computer Science
University of Oxford
ivan.flechais@cs.ox.ac.uk

Lizzie Coles-Kemp
Information Security Group
Royal Holloway University of London
lizzie.coles-kemp@rhul.ac.uk

Preface to the proceedings of Designing Interactive Secure Systems: Workshop at British HCI 2012, held at the University of Birmingham on 11th September 2012

British HCI,HCI 2012,Designing Interactive Secure Systems, DISS2012

1. INTRODUCTION

Welcome to the proceedings of the inaugural workshop on Designing Interactive Secure Systems (DISS 2012). This workshop was held in conjunction with the 26th BCS Interaction Specialist Group Conference on People and Computers at the University of Birmingham on September 11th 2012.

In recent years, the field of usable security has attracted researchers from HCI and Information Security, and led to a better understanding of the interplay between human factors and security mechanisms. Despite these advances, designing systems which are both secure in, and appropriate for, their contexts of use continues to frustrate both researchers and practitioners. One reason is a misunderstanding of the role that HCI can play in the design of secure systems. A number of eminent security researchers and practitioners continue to espouse the need to treat *people as the weakest link*, and encourage designers to build systems that *Homer Simpson* can use. Unfortunately, treating users as a problem can limit the opportunities for innovation when people are engaged as part of a solution. Similarly, while extreme characters (such as Homer) can be useful for envisaging different modes of interaction, when taken out of context they risk disenfranchising the very people the design is meant to support.

Better understanding the relationship between human factors and the design of secure systems is an important step forward, but many design research challenges still remain. There is growing evidence that HCI design artefacts can be effective at supporting secure system design, and that some alignment exists between HCI, security, and software engineering activities. However, more is needed to understand how broader insights from the interactive system design and user experience communities might also find traction in secure design practice. For these insights to lead to design practice innovation, we also need usability and security evaluation activities that better support interaction design, together with software tools that augment, rather than hinder, these design processes. Last, but not least, we need to share experiences and anecdotes about designing usable and secure systems, and reflect on the different ways of performing and evaluating secure interaction design research.

The objective of this workshop was to act as a forum for those interested in the design of interactive secure systems. By bringing together a like-minded community of researchers and practitioners, we aimed to share knowledge gleaned from recent research, as well as experiences designing secure and usable systems in practice. In doing so, this workshop became a crucible for building an interactive secure system design community, and forming collaborative partnerships to progress many of the aforementioned challenges.

Although this was the inaugural edition of this particular workshop, it built on the success of recent workshops in usable security at BCS HCI 2010 and BCS HCI 2011. This workshop also drew on the need highlighted at last year's NIST sponsored event on Security and Usable Security Aligned for Good Engineering for building bridges between different design disciplines, and providing a forum for sharing

anecdotal experiences about the design of usable and secure systems.

The aim of British HCI 2012 was to return to the conference's founding theme of *People and Computers*. Like the HCI field in general, the growing diversity of work in HCI-Security and Interactive System Design made the running of this workshop a timely opportunity to return to the design problems that motivated early work in these areas.

2. TECHNICAL PROGRAMME

In our call for papers, we invited 4-page position papers that would provoke discussion about the design of interactive secure systems. Our suggested list of topics included design techniques for socio-technical systems, technology for supporting interactive secure system design, usable and secure system evaluation, and experience reports.

The review process for each submission was rigorous. To select the technical programme, we were fortunate to be able to draw upon a panel of international experts; their expertise spanned the spectrum of interactive secure system design, from arts/design through to security engineering. Each paper submission received at least three reviews, although the vast majority received four. In addition to assessing the quality of work, reviewers were also asked to highlight areas they felt would provoke interesting discussion during the workshop itself. As such, these proceedings not only represent the efforts of the paper authors, but also the reviewers who provided detailed reviews and insightful suggestions to make sure authors get the most out of the DISS workshop experience.

3. ACKNOWLEDGEMENTS

This workshop would not have been possible without the hard work and dedication of our programme committee and external reviewers who, despite managing a slew of other commitments, still managed to find time to provide timely, high quality reviews to the paper authors. We are also grateful to the paper authors for contributing their work and participating in the workshop programme itself. Finally, we would like to thank the organisers of British HCI 2012 for hosting us, and the EU FP 7 *webinos* project for their sponsorship of this event.

4. ORGANISING AND PROGRAMME COMMITTEE

4.1. Organising Committee

Shamal Faily
Ivan Fléchais
Lizzie Coles-Kemp

4.2. Programme Committee

Yoko Akama
Henrik Artman
Debi Ashenden
Steffen Bartsch
Joel Brynielsson
Lynne Coventry
Paul Dunphy
Richard Ford
Peter Hall
Tristan Henderson
Christina Hochleitner
Marina Jirotka
Mike Just
Ronald Kainda
Linda Little
John Lyle
Vicki Moulder
Simon Parkin
Karen Renaud
Mary Theofanos
Joss Wright

4.3. External Reviewers

Joe Loughry
Fernando Muradas

The Effect of Users' Working Memory on Preference and Performance in Authentication Mechanisms

Marios Belk
University of Cyprus
1678 Nicosia, Cyprus
belk@cs.ucy.ac.cy

Christos Fidas
University of Cyprus
1678 Nicosia, Cyprus
christos.fidas@cs.ucy.ac.cy

Panagiotis Germanakos
University of Cyprus
1678 Nicosia, Cyprus
pgerman@cs.ucy.ac.cy

George Samaras
University of Cyprus
1678 Nicosia, Cyprus
cssamara@cs.ucy.ac.cy

An effective authentication mechanism should embrace both security and usability aspects as its purpose is to provide maximum protection of application providers' assets but as well usability and transparency to its end users, aiming to minimize cognitive overloads. With the aim to investigate the relation among users' working memory capacity and different types of authentication mechanisms, a study was conducted which entailed a psychometric-based survey for identifying users' working memory capacity, combined with a real usage scenario with two variations of authentication mechanisms. A total of 97 users participated in the reported study during a 5-month period providing interesting insights with respect to users' working memory and preference and performance of authentication mechanisms.

Authentication Mechanism. Working Memory. Usability. Preference. Performance.

1. INTRODUCTION

Research on authentication mechanisms has received significant attention lately with the aim to improve their usability and memorability, and at the same time decrease guessing attacks by malicious software and users (Inglesant and Sasse, 2010; Biddle et al., 2011). Researchers promote various designs of authentication mechanisms based on text and pictures, combinations of text and pictures, password managers and policies, etc. (Verma, 2012; Mihajlov and Jerman-Blazic, 2011; Biddle et al., 2011).

In this context, a large-scale study of half a million users, which investigated the password usage habits, supports the need of memorable and secure passwords (Florencio and Herley, 2007). A more recent study by Inglesant and Sasse (2010) that investigated the impact of password policies on users' productivity and experience, suggested that security policies should be driven by the users' needs helping them to set a stronger password instead of focusing on maximizing password strength.

Many shortcomings of authentication mechanisms arise from the limitations of human memory. The number of items the human brain can temporarily store is limited, with a short-term capacity (i.e., working memory) of ~3-7 items, depending on the task (Baddeley, 2007; Cowan, 2010). Enhanced working memory increases the connections and associations that can be built either between the items of the newly encountered information or between this information and information already stored in the long-term memory. Various research works (Cowan, 2010; Baddeley, 2007) argue that working memory has an effect on mental tasks, such as information processing, comprehension, learning, and problem solving.

Many studies indicate that working memory capacity varies among people and predicts individual differences in intellectual ability (Cowan, 2010; Baddeley, 2007). Such individual differences need to be further investigated aiming to understand whether they affect user interactions with authentication mechanisms.

In light of these challenges, this paper presents results of an empirical study which investigated the effect of working memory capacity of users towards preference and performance issues of two different types of authentication mechanisms; password and graphical authentication mechanisms.

2. METHOD OF STUDY

2.1 Procedure

A Web-based psychometric instrument was developed that assesses the capacity of the visuo-spatial sketchpad of users, which is the temporary storage mechanism responsible for processing visual and spatial information (Baddeley, 2007). This instrument aims to measure the amount of information the visuo-spatial sketchpad of a person can efficiently activate simultaneously by requesting from that person to memorize an abstract image and then compare that image with five other similar images.

Furthermore, a Web-based environment was developed for two introductory Computer Science university courses. Students were required to provide their demographic information during the enrolment process (i.e., email, age, gender, and department), and create their authentication key that was used for accessing the courses' material (i.e., course slides, homework exercises) and for viewing their grades. The type of authentication (password or graphical mechanism) was randomly provided during the enrolment process. At the end of the process the sample consisted of half of the students having enrolled with a password and the other half having enrolled with a graphical authentication mechanism. For the purpose of the experiment, in the middle of the semester, the system altered the students' authentication type; students that had enrolled with a password during the first half of the semester were prompted to create a new graphical authentication key and vice versa. The new authentication key would be used during the second half of the semester. The main aim of this process was to capture the interaction data of users for both types of authentication throughout the semester and further elicit their preference towards a particular type.

The password mechanism involved alphanumeric and special keyboard characters. A minimum of 6 characters including numbers, a mixture of lower- and upper-case letters, and special characters were required to be entered by the users. The graphical authentication mechanism involved single-object pictures with one-time authentication codes, where users had to select a minimum of 6 pictures (out of 30 available pictures) in a specific sequence, and was based on the recognition-based, graphical authentication mechanism proposed by Mihajlov and Jerman-Blazic (2011).

The total time required for successful authentication was monitored on the client-side utilizing a browser-based logging facility that started recording time as soon users entered the authentication Web-page, until they successfully completed the authentication process.

2.2 Hypotheses

The following null hypotheses were formulated: i) working memory capacity of users does not have a significant effect on users' preference towards password mechanisms or recognition-based, graphical authentication mechanisms, ii) there is no significant difference with regards to time needed to authenticate through a password mechanism or a recognition-based, graphical authentication mechanism among users having low, medium, and high working memory capacity.

2.3 Demographics of Participants

A total of 97 people participated in the study between January and May 2012. Participants varied from the age of 17 to 24, with a mean age of 20 and were undergraduate students of Electrical Engineering, Psychology and Social Science Departments. A total of 3461 successful authentications have been recorded during the 5 month period.

3. RESULTS

For our analysis, we separated the participants in three categories based on their working memory capacity: Low (N=27, f=27.8%, 33.37 average logins/user), Medium (N=50, f=51.5%, 38.88 average logins/user), and High (N=20, f=20.6%, 30.8 average logins/user).

3.1 Preference of User Authentication

An online questionnaire was provided to the students at the end of the study to express their preference towards a specific type of authentication (i.e., password or graphical). 66 out of the 97 students completed the questionnaire. In Table 1, we summarize the

preferences of authentication types according to the users' working memory capacity.

Table 1: *Users' Working Memory Capacity and Authentication Type Preference*

Working Memory Groups	Preference		Total
	Password	*Graphical*	
Low	*3*	*15*	*18*
Medium	*18*	*19*	*37*
High	*6*	*5*	*11*
Total	*27*	*39*	*66*

A Pearson's chi-square test was conducted to examine whether there is a relationship between users' working memory capacity and their preference towards a specific type of authentication mechanism (i.e., password or graphical). The results revealed that there is significant relationship between these two variables (Chi square value=6.139, df=2, p=0.046). In particular, examining each group (i.e., Low, Medium, High) individually with respect to preference towards a particular authentication mechanism, it has been identified that users with low working memory capacity significantly prefer graphical authentication mechanisms (Chi square value=8.000, df=1, p<0.01). In contrast, users having medium (Chi square value=0.27, df=1, p=0.869) and high working memory capacity (Chi square value=0.091, df=1, p=0.763) have not shown a clear preference towards a specific authentication mechanism.

3.2 Performance in User Authentication

A three by two way factorial analysis of variance (ANOVA) was conducted aiming to examine main effects between the users' working memory capacity (i.e., Low, Medium, High) and authentication type (i.e., password vs. graphical) over the time needed to access the system. Figure 1 illustrates the means of performance per working memory group and authentication type.

The analysis revealed that the main effect of users' working memory capacity on time needed to successfully authenticate is significant (F(2,97)=3.087, p=0.05). Furthermore, a pairwise comparison between the user groups and authentication types was conducted to examine whether they have a significant effect on the time required to authenticate to the Web-site.

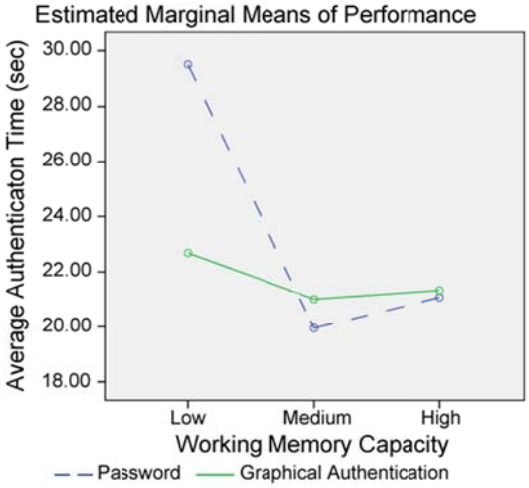

Figure 1: *Estimated Marginal Means of Performance per Working Memory Group and Authentication Type*

The results revealed that users of the Low working memory group performed significantly faster in graphical authentication (22.7 sec) than in password (29.5 sec) mechanisms (F(1,27)=10.630, p=0.003). However, users belonging to the Medium and High working memory group had no significant effect on performance between password and graphical authentication mechanisms (Medium Group: F(1,50)=0.263, p=0.611, and High Group: F(1,20)=0.006, p=0.939), as they performed almost equally the same in both authentication mechanisms compared to the Low working memory group (Figure 1).

4. CONCLUSIONS AND FUTURE WORK

The results presented in this paper constitute an effort to investigate whether individual differences of users have an effect on preference and performance issues related to authentication mechanisms. Such a research endeavour is based on the promise that understanding and modelling human behaviour with regards to security interactions can assist the design and deployment of more usable authentication mechanisms.

As a primary result, the study presented in this paper reveals a significant effect of users' working memory towards their preference and performance on authentication mechanisms. In particular, users belonging to the Low working memory group performed statistically significant faster with graphical authentication than a textual password mechanism. Similar results have been obtained relating with users' preference towards password or graphical

authentication mechanisms, as users belonging to the Low working memory group preferred graphical authentication mechanisms. In contrast, in the case of users that belong to the Medium and High working memory groups, no significant difference in preference and performance was observed between the two variations.

Taking into consideration that a graphical authentication mechanism is from a user's point of view a less demanding cognitive task than a password (recognition vs. recall of information), an interpretation of this result can be based on the fact that graphical authentication mechanisms leverage human memory for visual information (Biddle et al., 2011) and thus users with decreased working memory (i.e., Low) find graphical authentication mechanisms more usable and memorable since passwords are more demanding from a memory retrieval point of view. Another interpretation of this result can be based on the fact that users' preference towards graphical authentication might have also been affected by the picture superiority effect (Paivio and Csapo, 1973). On the other hand, users with enhanced working memory capacity could handle both authentication mechanisms more efficiently and effectively, hence no significant preference and performance were reported between the two types.

The limitations of the reported study are related to the fact that the participants were undergraduate students with an age between 17 to 24 years. Furthermore, carrying out a single assessment of users' working memory might not fully justify the users' classification into specific working memory groups. In this respect, further tests need to be conducted in order to reach more concrete conclusions. On the other hand, there has been an effort to increase ecological validity of the research since the participants were involved in real tasks, in their own physical environment, and without the intervention of any experimental equipment or observer.

Future research prospects include further investigating the effect of individual differences of users on preference and performance issues in security-related interactions (Belk et al., 2012). The overarching aim is to drive this research towards the development of a user-centred adaptation framework that will provide personalized security interactions based on cognitive factors of users.

5. ACKNOWLEDGEMENTS

We thank Dr. Artemios G. Voyiatzis for essential discussions related to usable security. The work is co-funded by the EU projects SocialRobot (285870) and CO-LIVING (60-61700-98-009).

6. REFERENCES

Baddeley, A. (2007) Working Memory, Thought, and Action. Oxford University Press, New York, NY, USA.

Belk, M., Fidas, C., Germanakos, P., Samaras, G. (2012) Do Cognitive Styles of Users Affect Preference and Performance related to CAPTCHA Challenges? In Extended Abstracts of CHI 12, Austin, TX, USA, May 5-10, 2012, pp. 1487-1492. ACM Press, New York, NY, USA.

Biddle, R., Chiasson, S., van Oorschot, P. (2011) Graphical Passwords: Learning from the First Twelve Years. ACM Security, Vol. 5, No. 4, pp. 1-43.

Cowan, N. (2010) The Magical Mystery Four: How is Working Memory Capacity Limited, and Why? Current Directions in Psychological Science, Vol. 19, No. 1, pp. 51–57.

Florencio, D., Herley, C. (2007) A Large-scale Study of Web Password Habits. In Proceedings of WWW 07, Banff, Alberta, Canada, May 8-12, 2007, pp. 657-666. ACM Press, New York, NY, USA.

Inglesant, P., Sasse, M.A. (2010) The True Cost of Unusable Password Policies: Password use in the Wild. In Proceedings of CHI 10, Atlanta, GA, USA, 10-15 April, 2010, pp. 383-392. ACM Press, New York, NY, USA.

Mihajlov, M., Jerman-Blazic, B. (2011) On Designing Usable and Secure Recognition-based Graphical Authentication Mechanisms. Elsevier Interacting with Computers, Vol. 23, No. 6, pp. 582-593.

Paivio, A., Csapo, K. (1973) Picture Superiority in Free Recall: Imagery or Dual Coding? Cognitive Psychology, Vol. 5, No. 2, pp. 176-206.

Verma, P. (2012) icAuth: Image-color based Authentication System. International Conference on Intelligent User Interfaces (IUI 2012), Lisbon, Portugal, February 14-17, 2012, pp. 329-330. ACM Press, New York, NY, USA.

Secure Communication in Disasters

Bangdao Chen Ivan Flechais Sadie Creese Michael Goldsmith A. W. Roscoe
Department of Computer Science
University of Oxford
[bangdao.chen; ivan.flechais; sadie.creese; michael.golsdmith; bill.roscoe]@cs.ox.ac.uk

This paper reports on the challenge of designing an application for bootstrapping secure communications in ad-hoc situations. The starting point of this work was based on prior work in "spontaneous security": making use of Human-Interactive Security Protocols (HISPs) which exploit a human-based unspoofable channel to bootstrap secure comunications. Our approach was to develop a realistic scenario in which spontaneous and secure communications are necessary, and to use this to drive the development of the application. We settled on exploring how to provide secure communications in disasters: situations where existing communication and security infrastructures may be unavailable. Using the disaster scenario to guide development, we implemented a mobile application which allows users to create ad-hoc WiFi networks and bootstrap secure communications over these networks.

Disaster, Spontaneous security, Human-Interactive Security Protocol

1. INTRODUCTION

Current security architectures are often rigid and unable to support situations where requirements are hard to identify upfront, or situations where flexibility is essential. Much of the domain of computer security is based on networks of shared secrets, or relies on certification structures and public key cryptography. It is notoriously difficult to make such structures dynamic, not least because the decisions about how parties are accredited and connections made are fixed by the designers of systems rather than those using them. Likewise, infrastructures like PKIs bind names to public keys, curtailing the possibility of naming flexibility. In response to these shortcomings, efforts have been made over the past 10 years to develop *spontaneous security*; one key aspect of this research has been the development of Human-Interactive Security Protocols (HISPs).

HISPs are protocols for authenticating systems and exchanging encryption keys based on the comparison of short strings over an *empirical channel*: an unspoofable human-based communication channel (Roscoe and Nguyen 2006). These protocols use a combination of digital communication channels and the empirical channel to bootstrap secure communications using knowledge that a human can only have gained via direct interaction with the system. The security of these protocols has been proven through formal evaluation; however, given the central role of a human agent in their correct use, the specific design and usability of their implementation is critical.

In order to design an application demonstrating the capabilities of HISPs and spontaneous security, we decided to use a realistic scenario as a means of exploring the problem domain. To represent the need for flexibility and security, we focussed on the challenges of communication in a disaster situation.

Disasters, like the Japanese tsunami of 2011, can destroy communication infrastructures as well as security infrastructures. In such events, establishing communications is a vital and time-critical job – one which cannot ignore the importance of security. Rescue operations often require cooperation among forces from different sectors, e.g. police forces, rescue teams, the military, as well as civilians. Security is necessary to guarantee the integrity, confidentiality and availability of communication channels in these operations, however it cannot be specified upfront. It is for these reasons that we decided to focus on this exemplar as a means of designing our application and showcasing the possibilities offered by spontaneous security.

In the following sections, we provide a brief overview of the security protocols that we use, and present our application – discussing the role of the disaster scenario in making specific design decisions.

2. SPONTANEOUS SECURITY

Spontaneous security aims to support users to decide *ad-hoc* on what systems can communicate with each other with no need for pre-existing

structures or infrastructures of encryption keys. A central means of achieving this is based on the discovery of highly efficient protocols for authenticating systems and exchanging keys, through the comparison of short strings generated by the parties involved.

A HISP allows two or more parties who trust one another, or a single party who trusts one or more others, to bootstrap a secure network using no more than an ability to communicate a small number of bits over an empirical channel. Another way of looking at them is that if the people involved create an insecure channel between their devices, and already have an unfakeable way of passing a small amount of information amongst them, then they can either turn the insecure channel into a secure one or discover the presence of an intruder who is trying to subvert it. The unfakeability of the empirical channel might arise from physical proximity (where participants can see and talk to one another), or real-time unspoofable interaction such as speaking on the phone.

The following describes the Symmetric HCBK protocol (SHCBK) (Roscoe and Nguyen 2008); a typical group HISP which can secure an arbitrarily-sized group.

1. $\forall A \longrightarrow_N \forall A' :\ A, \mathit{INFO}_A, hash(A, hk_A)$
2. $\forall A \longrightarrow_N \forall A' :\ hk_A$
3. users compare $digest(hk^*, \{\mathit{INFO}'_A | A \in G\})$, where hk^* is the XOR of all hk_A's for $A \in G$

Where $\longrightarrow_N$ represents a high bandwidth channel subject to the Dolev–Yao attack model (Dolev and Yao 1983), SHCBK has each node "publish" its name and a collection of information that it wishes to bind to that name. It also sends a hash[1] of a randomly generated key hk_A coupled with the name.

Once a node has received that information from all the others – and therefore become committed to the set of identities, *INFO* and hashed keys – it publishes its previously secret hk_A. The point is that by the time of this last publication, it is *committed* to all the data used in the above protocol, even though it does not yet *know* all the hk_As. A careful security analysis of this protocol (see (Roscoe and Nguyen 2008), for example) demonstrates that any attacker is unable to profit from combinatorial analysis aimed at getting the check-strings (i.e. digests) to agree even though nodes have different views of the authenticated information. Good HISPs such as SHCBK therefore offer maximum security for a given amount of human effort.

[1] Hash means a standard cryptographic hash function that has two main properties: collision resistance, and inversion resistance.

The digest function (Roscoe and Nguyen 2006, 2008) is designed so that, as hk varies, the probability that $digest(hk, X) = digest(hk, Y)$ for $X \neq Y$ is less than ϵ, where typically ϵ is very close to the theoretically optimal value of 2^{-b} for b the number of bits in the output of $digest$. It must also have the property that for any fixed value d, the chance that $digest(hk, X) = d$ as hk varies is less than ϵ also. More details of this protocol can be found in (Chen et al. 2012).

As an example, Alice and Bob want to create a secure connection between their mobile devices. First they connect their devices using a normal channel (e.g. Bluetooth, WiFi), and we assume this connection is initially insecure. Alice and Bob then run the protocol which generates a check-string (e.g. a six-digit number) and exchange this value. One method of exchanging the value is for Alice to read the check-string and Bob to type it on his own mobile device. Another is for them to both read it, compare the one heard with the check-string displayed on their own devices, and confirm that it indeed matches.

Because Alice and Bob are in physical proximity, they can make sure that no one can fake this check-string. The communication channel they use to exchange the check-string is an empirical channel: no one can alter the check-string and no one can fake the origin of the check-string. The check-string is then used to test the presence of the attacker. For example, if the check-string entered matches the one generated by their own devices, they can be confident that the connection has been secured; otherwise, it is likely that the connection is insecure and an attacker (e.g. a man-in-the-middle attacker) is in presence.

3. HISP APPLICATION

Based on the disaster scenario, we assume that people will have their own mobile devices to communicate, and have designed our application accordingly. The connection between devices uses the peer-to-peer (ad hoc) WiFi mode. This allows the creation of a communication network for situations where the existing infrastructure may be unavailable. To secure communications over this network we break down the application into three steps: group formation, running the protocol, and ongoing communication.

To simplify our discussion, an important assumption has to be made before bootstrapping security for a group: members of a group are capable of verifying the legitimacy of every other member of the group.

3.1. Group formation

Formally, bootstrapping a group can be defined as follows: all members acknowledge a list L, which

contains details of all members; the resulting group G contains exactly the same number of members recorded in L and no one, except for the members included in L, can be allowed to join G. To satisfy this task, we need to identify and overcome the following challenges: collecting members' information and counting the number of members.

In GAnGs (Chen et al. 2008) the authors present two solutions for collecting group members' information when they are in the same room. The first solution is to use an untrusted projector as a central node by displaying its Bluetooth address as a 2D barcode. All members connect their mobile phones to the projector by reading this barcode and send their details to the projector which then broadcasts the list L to the group; whilst this is fairly simple, it is not suitable for the disaster scenario. The second solution is to create a tree structure to collect members' information one by one by reading 2D barcodes of Bluetooth addresses. This can be a laborious process which involves a large amount of human effort (e.g. 30 human interactions for a group of ten members).

Our solution is tailored to WiFi, where one device (the Initiator) can broadcast[2] messages include its IP address and profile. Other devices within communication range can pick up these messages and display the Initiator's profile. The user can then choose whether or not to join this group.

The second step is to count group members – verifying that the size of the group G matches the size of list L. This is to prevent an insider from creating multiple fake identities and passing the authentication step. For example, when comparing digests, an insider can compare digests multiple times and fake the presence of the identities he/she has created.

When a group is small (e.g. 3 or 4 members), counting is straightforward. However, as the size of the group grows, so does the likelihood of mistakes. For example, in a group of over 100 members, it is unreasonable to expect all members to count the whole group consistently and individually. One simple solution is to utilise "crowd-knowledge" to remove illegal members from list L. Another is to divide a large group into small groups, and create a strategy for merging the smaller groups into a larger one.

In our application, we assume that the Initiator manages the counting on behalf of others. This means that the Initiator knows how many members are in the group and he/she can remove illegal identities.

[2]This can be achieved by sending UDP packets to the broadcast address. The broadcast address is computed by the following equation: Broadcast address = IP address | ~Netmask.

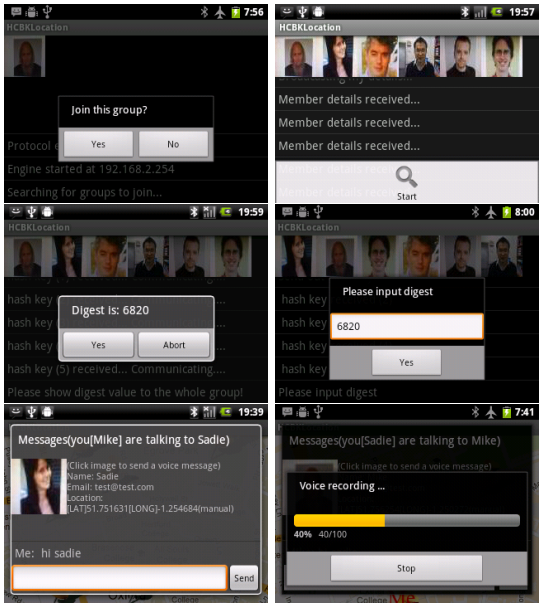

Figure 1: *Screen shots (left to right, top to bottom): (i) join a group; (ii) start the protocol; (iii) display the digest; (iv) enter the digest; (v) send a text message; (vi) send a voice message.*

3.2. Running the protocol

Since the Initiator manages the group formation, they are also responsible for starting the HISP. When the protocol starts, devices broadcast Message One: $A, \mathit{INFO}_A, hash(A, hk_A)$. At this stage, each device must ensure that it has received exactly N unique copies of Message One, where N equals to the size of the group. This is critical to guarantee that all members are committed to hk_As.

A device must not process other messages until it has received all N copies of Message One. This will guarantee the failure of this stage will be detected in the digest comparison stage.

When all devices have received all Message Ones, they start to broadcast Message Two: hk_A. Similar to the previous stage, all devices need to make sure they have received all N copies of Message Two. Once they have, they compute and display the digest value.

The digest comparison stage serves to check whether there is any mistake or attacker in the previous two stages of communication. To symmetrically compare digests in a group of size N ($N > 2$), a total number of $N(N-1)/2$ two way interactions, or $N(N-1)$ one way interactions (for example, taking photos of 2D barcodes), have to be made. Given $N = 8$, there can be 28 to 56 interactions. In order to improve usability, we must reduce the number of human interactions involved in this stage.

Figure 2: *Devices running the secure communication application.*

In our application, we have assumed that there is one trustworthy member who is known by the rest of the group, and all members are close to each other so they can hear each other speaking. We further assume that the trustworthy member is the Initiator.

The Initiator reads out the digest to the group members who enter this value into their own devices (manually entering the digest has already been shown to be both usable and secure (R. Kainda and Roscoe 2009)). The device then checks that the value entered matches with its own version. When this is completed, all members raise their hands to indicate that the digest comparison is successful; or speak loudly if the digest comparison has failed. The Initiator reading the digest to the rest of the group is a typical one-to-many empirical channel. Since we assume the Initiator is trustworthy, we have used this empirical channel to improve usability; if we consider the step of raising hand and checking as one single interaction, the total number of interactions is reduced to $2N$.

3.3. Communication: cooperation in disasters

After the protocol is successful, we use Diffie-Hellman public keys (included in $INFO_A$) to generate shared symmetric keys and then we generate a group key by using the symmetric keys. We have implemented text messaging, voice messaging and sending photos on the basis of a map function (see Figure 2). We have also augmented text and voice communication with location and photography functions to facilitate rescue operations. The location function can track a user's location on the map; the photo function is used to post notices on the map (e.g. newly hazardous areas).

4. FUTURE RESEARCH

We have discussed basic issues of group formation and comparison of digests. More needs to be done to explore how to establish and maintain a sufficiently large ad hoc group of devices using the connection technology available in a given scenario. For example, should a group be formed using a single initiator, a tree structure, broadcasting over a fully connected graph, or some other topology? This question can be posed both for initial network formation and for digest comparison between humans. How should we manage group amalgamation and splitting, or adding and removing single members? We have also indicated that naming is difficult in disasters, but how to manage identities remains a significant challenge in disasters.

5. CONCLUSION

We have discussed how we designed and implemented a secure communication application in a disaster scenario. Our implementation shows that we can optimise performance by designing and regulating the group formation and the digest comparison process. It also shows that we can easily build various communication functions on top of this security platform to facilitate rescue operations in disasters.

REFERENCES

Chen, B., Nguyen, L., and Roscoe, A. (2012). Reverse authentication in financial transactions and identity management. *Mobile Networks and Applications*, pages 1–16. 10.1007/s11036-012-0366-2.

Chen, C.-H. O., Chen, C.-W., Kuo, C., Lai, Y.-H., McCune, J. M., Studer, A., Perrig, A., Yang, B.-Y., and Wu, T.-C. (2008). Gangs: gather, authenticate 'n group securely. In *Proceedings of the 14th ACM international conference on Mobile computing and networking*, MobiCom '08, pages 92–103, New York, NY, USA. ACM.

Dolev, D. and Yao, A. (1983). On the security of public key protocols. *Information Theory, IEEE Transactions on*, 29(2):198 – 208.

R. Kainda, I. F. and Roscoe, A. (2009). Usability and security of out-of-bound channels in secure device pairing protocols.

Roscoe, A. W. and Nguyen, L. H. (2006). Efficient group authentication protocols based on human interaction. In *Proceedings of ARSPA 2006*.

Roscoe, A. W. and Nguyen, L. H. (2008). Authenticating ad hoc networks by comparison of short digests. *Information and Computation*, 206:250–271.

Software for Interactive Secure Systems Design: Lessons Learned Developing and Applying CAIRIS

Shamal Faily
Department of Computer Science
University of Oxford
shamal.faily@cs.ox.ac.uk

Ivan Fléchais
Department of Computer Science
University of Oxford
ivan.flechais@cs.ox.ac.uk

As systems become more complex, the potential for security vulnerabilities being introduced increases. If we are to provide assurances about systems we design then we need the means of analysing, managing, and generally making sense of the data that contributes to the design. Unfortunately, despite ongoing research into tools for supporting secure software development, there are few examples of how tools can be used to help build and support design models associated with security and usability. This paper summarises some of our experiences developing and applying CAIRIS: a requirements management tool for usable and secure system design. We describe our motivation for building CAIRIS, summarise how it was built and evaluated, and present our experiences applying it to real world case studies.

CAIRIS,Requirements Management,Security,Usability

1. INTRODUCTION

As systems become more complex, the potential for security vulnerabilities being introduced increases. This means that if we are to provide any assurances about systems that we design then we need some means for analysing, managing, and generally making sense of all the data that contributes to a system's design to ensure such vulnerabilities are not unintentionally introduced. While there has been ongoing research into software tools to support the development of secure software, there has been comparatively little work on tools for reasoning about security and usability models. Without this ability, it is difficult to predict the usability implications of security design decisions and vice-versa, the security implications of usability decisions; this becomes particularly difficult when considering how these implications might change in different contexts of use.

To help understand how software tools can support security and usability design techniques, we developed CAIRIS (Computer Aided Integration of Requirements and Information Security): a requirements management tool for secure and usable system design. Since developing the initial prototype in 2009, we have evolved CAIRIS based on our experiences in several real-world case studies. In this paper, we reflect on some of these experiences building and applying CAIRIS. In Section 2, we briefly describe some of the challenges that motivated our approach to designing CAIRIS, before summarising how the tool has been developed and evaluated in Section 3. In Section 4, we discuss some of our experiences and problems faced in using and maintaining CAIRIS.

2. RELATED WORK

Because *requirements* are a recognised boundary object across security, usability, and software engineering models, requirements management tools have been proposed as a basis for supporting security and usability design activities. Their potential for extensibility is illustrated by the DOORS requirements management tool (IBM 2010) which, with the aid of its DXL scripting language, supports extensions for specifying positive and negative scenarios of user behaviour(Alexander 2002). However, the lack of distinct semantics for the underlying concepts associated with these techniques means that analysts need to manually maintain links between requirements and non-requirements artifacts.

By structuring the data being managed according to a specific meta-model, model-based approaches

Proceedings of BCS HCI 2012 Workshops
Designing Interactive Secure Systems

address this traceability management problem. However, modelling languages and tools tend to consider requirements only as a notational concept. For example, while UMLSec—a UML profile for secure system development(Jürjens 2005)—supports the concept of a *security requirement*, this is depicted only as a UML stereotype. Rather than being concerned with how these might be analysed, the notation is concerned with the requirement's deployment rather than its specification.

Further problems arise when trying to integrate tools grounded in similar, but subtly different, conceptual models. For example, if we assert that a misuse case *threatens* a use case, do we agree what it means for the use case to be threatened? Does the misuse case threaten the work carried out by the use case, or the assets associated with it? Houmb et. al (Houmb et al. 2010) faced some of these problems when integrating tools based on different techniques. Their experiences indicate that while building heuristics into tools to help with integration is useful, these alone can't replace the expertise needed to apply the techniques themselves. Consequently, while integrating tools and concepts can help verify requirements, few tools provide support for eliciting or validating them.

3. BUILDING AND EVALUATING CAIRIS

Based both on the IRIS meta-model (Faily and Fléchais 2010)—which characterised our ideas about how concepts from requirements, security, and usability engineering might interrelate—and lessons learned from existing tools, we designed and developed CAIRIS to appeal to the following design principles:

- Familiarity: The tool itself should not add to pre-existing cognitive burdens; given the difficulty associated with grasping new concepts and learning new notations, the tool and its artifacts should require no more cognitive overhead than learning how to use the techniques associated with IRIS meta-model.
- Extensibility: Because the tool was to be used in several case studies, new insights might arise from its use; this could include identifying unnecessary functions, or the need for new functionality. Moreover, it should be possible to quickly modify the tool to implement the suggested changes and assess their impact during, or shortly after, an intervention.
- Standardisation: As well as structuring the collected data, we wanted the tool to be used to support a variety of existing analysis techniques. Crucially, we wanted to ensure that the tool supported each without changing standard concepts or the manner in which each technique normally operated. This meant that different people might use the tool to support different techniques, according to their expertise and responsibilities. Consequently, given that the meta-model allowed data collected through one technique to inform another, traceability between model concepts needed to be automatic.

We developed CAIRIS using a prototyping approach, over five iterations.

In the first iteration, CAIRIS was developed in parallel with the IRIS meta-model. The tool was used to elicit data using contemporary examples where multiple contexts of use were evident. One of these examples involved analysing contemporary news reports and documentation about the Vélib bicycle sharing system to elicit security requirements which would not compromise the usability of Vélib. The objective of this phase was to determine whether the concepts necessary to model the different problems were reflected in the IRIS meta-model.

Based on early feedback from the Requirements Engineering community (Faily and Fléchais 2009), additional concepts were added to the IRIS meta-model and the tool was evolved to support these. As the meta-model became more elaborate, additional model views were incorporated into the tool, and the architecture was re-factored to allow scaleability should further model elements and associations need to be added.

In the second iteration, we created a specification exemplar based on the NeuroGrid e-science project (Geddes et al 2006) to validate whether the tool was capable of modelling a complete, non-trivial problem. Using CAIRIS, the resulting NeuroGrid model was validated with one of the previous project stakeholders.

The final three iterations involved applying CAIRIS in three separate case studies; these studies are described in more detail in (Faily 2011b)

CAIRIS was written primarily in Python, and used the open-source wxPython and pyGTK frameworks for windowing and visualisation support, and NumPy for matrix manipulation. MySQL was used for management and access to model data. Although CAIRIS is primarily maintained by members of the Security research group at the University of Oxford, it has been released to github as an open-source project under an Apache Software license.

4. EXPERIENCES

Space constraints mean we are unable to reflect on all of our experiences with CAIRIS. We do, however, highlight three particular experiences that, we believe, might provide useful insights to designers of future software tools for secure and usable system design.

4.1. Building on similarities rather than differences

One of the first challenges we needed to address was reconciling the several different security requirements engineering meta-models that had previously been proposed; this would be particularly challenging given the lack of consensus about what a security requirement is, e.g. (Tøndel et al. 2008). Rather than trying to tease out lessons learned from all these models, we instead decided to select a particular model and attempt to scale this given the additional usability elements we wanted to consider. We chose the Information Systems Security Risk Management (ISSRM) modelling language Mayer (2009) because not only was it largely orthogonal to usability, it also attempted to align with goal-oriented requirements languages. While the concepts associated with those languages were incompatible with IRIS, we believe that on-going research in both CAIRIS and their languages might lead to synergies in the future.

Our approach came at the cost of simplifying the risk analysis approach adopted by ISSRM. This ruled out the ability for modelling sophisticated risk analysis strategies, such as blended threats where an attacker might exploit two seemingly innocuous vulnerabilities to achieve catastrophic results. Nonetheless, by keeping the IRIS meta-model as simple as possible without compromising any of the fundamental concepts, we were able to progressively incorporate concepts over time.

For example, based on what was originally an unconnected research finding about the usefulness of argumentation models for supporting persona development Faily and Fléchais (2011), we were eventually able to align concepts from IRIS with what were incompatible goal-oriented requirements techniques; this is described in more detail in Faily (2011a). We believe that the success of this approach relied not on seeing how social-goal models could be built into CAIRIS, but on how CAIRIS could add value to complementary tools which are better suited to analysing these in more detail.

4.2. Specifying contexts of use

From the outset, the concept of an *environment* was supported in CAIRIS; this was introduced to make it possible to specify and reason about the elements of several different contexts of use for a given system. However, based on the results of our case studies, the results of trying to specify formal contexts of use were mixed.

While environments could be of any type, those of most value tended to be social or cultural rather than physical. When considering one example based on the NeuroGrid exemplar, we noticed that security properties associated with certain assets had markedly different security properties in different environments. In the three case studies, the security properties varied less, and the environments were used to compartmentalise the analysis of activities according to the context of most relevance. Occasionally, however, some discussion arose by comparing the same tasks carried out in different social contexts, or discussing how the tasks carried out in one environment had an impact in others.

We were also interested in how variations of context could be composed to introduce new contexts of use. Individually, such environments might be innocuous but, when combined, new phenomena might be observed that might not otherwise be seen in the separate models. To investigate this, CAIRIS allowed composition of an environment based on one or more other environments. In practice, however, this did not prove to be very useful. Although the environments modelled in the case study examples were non-trivial, they were also distinct enough that combining them added little value to the analysis carried out. However, reasoning about dependencies between two environments was occasionally useful when considering how attackers might exploit knowledge about one environmental context to cause a shift from one environment to another, or how a risk in one context lead to a subsequent risk being introduced in another.

4.3. Using CAIRIS

CAIRIS was not designed to be a general purpose tool. One of the initial motivations for building the tool was to support participative workshops where stakeholders could discuss different models, and examine the impact of model changes in real time. However, when CAIRIS was used for supporting design activities for the EU FP 7 webinos project, we faced two new challenges. First, many of the CAIRIS users didn't have the necessary technical background to install and setup CAIRIS. Second, the users were only knowledgeable in some of the capabilities of CAIRIS. As a result, of the users

that did install and use CAIRIS, all used CAIRIS for little more than a tool for specifying and managing personas.

Because there was little time available on the project to run training sessions, we used the project wiki to capture structured data that could then be imported into CAIRIS. We created structured page templates for design artifacts like scenarios, use cases, and personas. We then created scripts that could be used to convert this content into compatible XML models that could be imported into CAIRIS. In addition to this, we provided guidance and support for the rest of the project on the use of the templates. Eventually, this approach was extended to security and requirements models as well. As a result, although the wiki was still used to browse data, most of the webinos security, usability, and requirements model was stored as text in a project git repository, with a *build* script used to create a consolidated CAIRIS model on demand. More details about this process can be found in Faily et al. (2012).

5. CONCLUSION

In this paper, we described some of our experiences in designing, building, and using the CAIRIS requirements management tool. In providing our candid experiences with CAIRIS, we aim to begin filling the hitherto unnoticed gap in the literature on tools for secure and usable system design.

While the tool was designed to support only a single researcher (the main author), the CAIRIS user community is slowly beginning to grow. Following the introduction of the tool to the Oxford Software Engineering Programme's *Design for Security* course, practitioners are beginning to use CAIRIS to address their own security design challenges. As industrial take-up grows, we plan to evaluate how well CAIRIS, and software tools in general, tackle the security usability design problems practitioners currently face.

6. ACKNOWLEDGEMENTS

The research described in this paper was funded by EPSRC CASE Studentship R07437/CN001, and the EU FP7 webinos project (FP7-ICT-2009-05 Objective 1.2).

REFERENCES

Alexander, I. (2002). Initial industrial experience of misuse cases in trade-off analysis. In *Proceedings of the IEEE International Requirements Engineering Conference*, pages 61–68. IEEE Computer Society.

Faily, S. (2011a). Bridging User-Centered Design and Requirements Engineering with GRL and Persona Cases. In *Proceedings of the 5th International i* Workshop*, pages 114–119. CEUR Workshop Proceedings.

Faily, S. (2011b). *A framework for usable and secure system design*. PhD thesis, University of Oxford.

Faily, S. and Fléchais, I. (2009). Context-Sensitive Requirements and Risk Management with IRIS. In *Proceedings of the 17th IEEE International Requirements Engineering Conference*, pages 379–380. IEEE Computer Society.

Faily, S. and Fléchais, I. (2010). A Meta-Model for Usable Secure Requirements Engineering. In *Proceedings of the 6th International Workshop on Software Engineering for Secure Systems*, pages 126–135. IEEE Computer Society.

Faily, S. and Fléchais, I. (2011). Persona cases: a technique for grounding personas. In *Proceedings of the 29th international conference on Human factors in computing systems*, pages 2267–2270. ACM.

Faily, S., Lyle, J., Paul, A., Atzeni, A., Blomme, D., Desruelle, H., and Bangalore, K. (2012). Requirements sensemaking using concept maps. In *Proceedings of the 4th International Conference on Human-Centered Software Engineering*. Springer. To Appear.

Geddes et al (2006). The challenges of developing a collaborative data and compute grid for neurosciences. *Computer-Based Medical Systems, 2006. CBMS 2006. 19th IEEE International Symposium on*, pages 81–86.

Houmb, S. H., Islam, S., Knauss, E., Jürjens, J., and Schneider, K. (2010). Eliciting security requirements and tracing them to design: an integration of Common Criteria, heuristics, and UMLsec. *Requirements Engineering*, 15(1):63–93.

IBM (2010). IBM Rational DOORS.

Jürjens, J. (2005). *Secure systems development with UML*. Springer, Berlin.

Mayer, N. (2009). *Model-based Management of Information System Security Risk*. PhD thesis, University of Namur.

Tøndel, I. A., Jaatun, M. G., and Meland, P. H. (2008). Security requirements for the rest of us: A survey. *IEEE Software*, 25(1):20–27.

Community-centric engagement: lessons learned from privacy awareness intervention design

Lizzie Coles-Kemp
Information Security Group
Royal Holloway
Egham TW20 0EX
lizzie.coles-kemp@rhul.ac.uk

Debi Ashenden
Informatics and Systems Engineering
Cranfield University
Shrivenham SN6 8LA
d.m.ashenden@cranfield.ac.uk

Visualisation and Other Methods of Expression (VOME) was a research project with the main objective of developing methods of expressing a wide range of on-line privacy-related concepts. Having built this wider vocabulary, VOME was also tasked with developing a range of privacy awareness interventions for both on and off-line contexts. Examples of VOME interventions include a card trading game, participatory video and embedded on-line interaction tools. In order to develop these interventions, VOME conducted a range of user-centric and participatory design engagements in communities across the UK. Engagements included: think-out-loud technology evaluations, participatory video development, "show and tell" family workshops, participatory theatre and community art collage building. Within each of these engagements, the project used qualitative social research to ground the design of its interventions. The communities in which VOME engaged represented a broad range of social and economic demographics and demonstrated varying levels of digital literacy. During its three and half years of engagement, VOME built up a rich set of lessons learned from using such a wide range of engagement approaches. The lessons learned address a number of key topics: tension of engagement control between participants and researchers, importance of appropriately situating the engagement and limitation of participant segmentation. This short paper outlines VOME's key findings within each of these topics.

Key words: Engagement, intervention, digital literacy, privacy awareness.

1. INTRODUCTION

Visualisation and Other Methods of Expression (VOME) is an interdisciplinary project that focuses on questions related to privacy, identity and consent in on-line services. The research is used to ground the development of technology designed to support privacy and consent decision-making. Examples of VOME interventions include a card trading game, participatory video and embedded on-line interaction tools, for example interactive information flow maps. From the start, VOME set out to engage with citizens who, to date, had not been included in privacy design studies. Sunderland City Council gives the example of one such group of citizens:

> "Consider the case of a young person, aged 13 years, who is at risk of offending. They are one of the target groups for the Empowering Young People programme which will use on-line service delivery as part of the method of delivery. Some of their peers have heard about the scheme and are saying it will be used by the police to keep track of them. Our young person is shy, reserved and has some learning difficulties. How will they express their concerns about how their data will be used, or will they simply choose not to engage? How might this be further complicated if the young person were to be a member of a minority ethnic group? Would a set of tools designed for youth work support workers help clarify the issue, and engage the young person? How might the youngster explain how the scheme safeguards information rights to their peers?" Conn Crawford, Sunderland City Council.

The project teams conducted a number of qualitative social research studies that required researchers to explore how participants engage with technology and the decision-making logic in operation when personal information is disclosed.

Proceedings of BCS HCI 2012 Workshops
Designing Interactive Secure Systems

The research teams realised quite quickly that traditional qualitative methods had limitations with the groups with which the project wanted to engage. In particular, researchers learned that unintentionally the traditional data gathering and engagement approaches were not inclusive and partially or completely excluded those with lower levels of digital and visual literacy and those with lower levels of social confidence. For example, our mixed method studies which used light technological evaluation, observations, interviews and surveys also proved challenging for those with lower levels of digital and visual literacy (Coles-Kemp and Zabihi, 2011). This realisation both influenced the research approaches taken and method and media for data gathering.

2. APPROACH

In VOME, the traditional approaches found in security and privacy technology design were supplemented with more inclusive data gathering approaches. These approaches include: think-out-loud technology evaluations, participatory video development, "show and tell" family workshops, participatory theatre and community art collage building. Using these approaches also taught the researchers about the need to cede control to participants and carefully segment communities into different constituencies. These approaches also clearly showed the need to design culturally sympathetic research approaches and to carefully select the media for data collection.

2.1 Participatory Video

VOME supported Hudson's Youth Project in the London borough of Newham to produce two videos – one a documentary video that the VOME project specifically commissioned and the other a music video that the young people asked if they could make. The aim was to create a participatory video where the young people determined what was filmed, how it was produced and, most importantly, the message conveyed. With the documentary video researchers were part of the focus groups that were run to stimulate discussion and the ensuing video displays strong characteristics of researcher involvement. The music video on the other hand required no input at all from the researchers and used a medium that the young people were more comfortable with. The music video, 'Internet Saint, or Online Demon?', was a more effective approach for gathering young people's views as it was made without intervention from researchers. This result showed that the design approach needs to use a medium for data gathering that is culturally sympathetic to the target demographic.

2.2 Think-Out-Loud Technology Evaluations

VOME used think-out-loud technology evaluations to better understand the logic behind personal information disclosure in different situations. For example, researchers developed an on-line registration prototype which represents a mock-up council, named Your Local Council (YLC), which offers an on-line smartcard registration service. This is a situation that many of the participants were familiar with and was situated in a relationship with an institution that many of them had. In the research activity, after the introduction to smartcard services provided by YLC, users were asked to interact with the website which guides them through a sequential registration process that consisted of the following pages:

(i) Introduction to smartcard services and selection of services - The user starts by selecting one or more services.
(ii) About us
(iii) Personal Information Requirement - The service provider informs the user that certain personal information is required to use the service, what will happen to the data as well as the reason for collecting it. This information will be communicated to the user with help of an interactive dataflow diagram that displays who has access to what type of personal information.
(iv) Service Agreement - The service provider gives a contract of their agreement with the user to keep as a reference. It is an overview of selected services, privacy policy and terms & condition.
(v) Registration - The final step of the process is a registration page that displays a form where the user discloses the necessary personal information for acquiring the selected service(s).

Participants were exposed to three privacy interventions during the registration process. Privacy awareness was measured using a mixed-methods approach in three parts:

- Questionnaire to categorise the participants in terms of their digital literacy.
- Engagement with digital probes (YLC website) to encourage reflection (captured through think-out-loud)
- Interview to evaluate levels of privacy awareness with the participant.

2.3 "Show and Tell" Family Workshops

One drawback to the think-out-loud technology evaluation in the previous section is that it constrained participants to reflect on a particular type of technology in a particular context.

As a result, VOME ran a number of "show and tell" workshops where a small group of participants, matching a particular set of criteria, would spend time showing and reflecting how they used ICT within the family setting and talking about the type of personal information that they disclosed in the process. In this approach, in contrast to the think-out-loud technology evaluations, participants situate their technology use in a context to which they can relate. Participants also shape the flow of the engagement by bringing up points for discussion from the reflection sessions.

One example of such a workshop took place in northern England in July 2011. The workshop involved six granddaughters (GDs) and six grandmothers (GMs). It was staged in the part of a northern English town where local granddaughter/grandmother pairs could be recruited through a community center. Preliminary work had already identified that there were close family pairings and internet active family members who met the criteria for participation.

The GMs were aged 55+ and four of the six GMs were great-grandmothers. They included a mix of active social networkers and those without accounts for any social networks. One GM used social networking to keep contact with relatives in Australia. Two of the others used the Internet for email and on-line shopping. The non-social networkers all had experience of family members who use social networks. The event was facilitated by a community leader and a VOME researcher. The workshop began with an introduction to VOME's research and the process for the day. This explained that VOME's work is on personal information control, but didn't develop the theme. The "show and tell" session in this instance used GMs and GDs in a pair. Each GD/GM pair used the big screen and the computer in the workshop room to show their activities. The participant group was then encouraged to discuss issues that arose and write down thoughts and reflections. The follow-on "show and tell" session picked up on the themes identified in the reflection session.

2.4 Participatory Theatre and Community Art

Whilst "show and tell" family workshops moved VOME researchers some way towards participant-led situated research, researcher intervention was still present and for some participants that presence was a barrier to engagement with the research. As a result, VOME pushed further towards participant-led research engagements by using participatory theatre and community collage making. This form of engagement would often involve collaboration with performance artists, in particular clowns, who initiate engagement with members of the public and introduce them to some very simple research tasks.

One example of this is collage building in Middlesbrough Railway Station. The focus of this activity was to ask members of the public what type of secrets they keep on-line and what secrets they try to find out on-line. Performance artists initially engaged with participants. Those who then wanted to answer research questions were invited to produce some drawings, text or sound recordings for a collage that was built in the underpass of Middlesbrough railway station. The same approach was used across four community centres, the central library and the city museum in Sunderland. This engagement approach had the least direct input from researchers in terms of participant engagement but had considerable input from the researchers in the setup, observational work while the engagement was on-going and post engagement analysis.

3. LESSONS LEARNED

Lessons learned from developing the engagement programme can be grouped into three areas: control tension, situation and segmentation limitation.

3.1 Control Tension

When designing engagements, a tension emerged between the control the researcher needed to have over the engagement and the control that the researcher needed to cede to the participants.
During the initial engagements such as the participatory video and think-out-loud technology evaluations, the need to give control to participants (Ospina et al, 2008) became clear. This means that engagement cannot start from a fixed question, but that researchers have to interpret what is produced during the engagement. Researchers also learned that even light control might be too dominant for some groups of participants. Engagement with communities is often mediated through powerful stakeholders such as community and youth workers. Therefore, in terms of control, it is also important to understand the involvement and influence of stakeholders close to the end users. This is often unavoidable but, as French et al (2010) point out, such stakeholder influence needs to be acknowledged in any engagement activity as it will have an impact on how messages are received.

The documentary video made by Hudsons Youth Project demonstrated too much control by researchers as researchers' words were repeated back through the video suggesting that what was presented was what the young people believed the

researchers wanted to hear rather than what they actually felt.

In order to cede control to participants, design researchers need to reduce the influence of researchers and gatekeepers as far as possible and select media for data collection that is culturally sympathetic to the target demographic.

3.2 Situation

Engagement fails when it assumes a transmission model of communication, ignores lay or public perceptions and when it presents technological fait d'accompli. The need to develop insight is emphasised by French et al (2011) and the key lesson to take away is that for engagement to be successful it is vital to understand end users and the contexts that determine their attitudes and behaviours from their point of view – not how the researcher imagines it to be.

In the think-out-loud technology evaluation, the context of registering with an on-line Council service did not resonate with all the participants. Also, those evaluations were run in a community or UK online centre and this was not a physical space that all the participants would use for on-line registration. Another situational issue was the lack of family context. Many of the less digitally capable participants would typically register for on-line services with the help of family members. In this case, on-line engagement without other family members present as well as outside the home was not natural. As a result, fewer participants with this type of background took part in the think-out-loud technology evaluation and for those that did; their evaluation was stilted because of their lack of familiarity with the context.

Therefore, design researchers need to recognise the importance of appropriately situating the engagement in as natural a setting as possible. This is often referred to as 'ecological validity' (Hayes, 2000, p.105) and is often absent in laboratory-based studies.

3.3 Limitations of Segmentation

As was seen with the participatory video, the manner in which an engagement is responded to depends on the segmentation of the participants. Also, there will be a range of responses within each segment. In particular, issues of digital literacy, degree of stability in home and social lives, cognitive capacity and degree of social confidence were some of the dominant factors in influencing segmentation response. It is therefore important that design researchers do not present the experiences of demographic segments as a homogeneous group with a single consistent attitude towards technology and risk. Otherwise the research stands to unknowingly alienate an often significant proportion of the user community. More subtle approaches to segmentation are needed. It is also important for researchers to recognise that different research focuses need different types of segmentation. In the VOME project, we found that community-based segmentation had an important role to play (Andreason, 2006) and therefore segmentation according to family settings was important.

4. CONCLUSION

The push towards participant- controlled engagement was a valuable journey. As intervention designers, we were able to gather the views and inputs of a wider audience. By not taking account of the lessons learned systems are likely to be designed that end users will resist using either because they don't take sufficient account of existing attitudes and behaviours or because they do not accord with how systems are used in a real-world setting. The lessons learned will influence the methodologies used in future studies around not just privacy awareness but online awareness and behaviour in general.

5. REFERENCES

Andreason, A. (2006) Social Marketing in the 21st Century. Sage, London, UK

Coles-Kemp, L., Kani-Zabihi, E. (2011) Practice Makes Perfect: Motivating confident on-line privacy protection practices. IEEE International on Social Computing, 9 - 11 October, Anonymous IEEE, pp. 866-872

French, J., Merritt, R., Reynolds, L. (2011) Social Marketing Casebook. Sage, London, UK

Hayes, N. (2000) Doing Psychological Research Open University Press, Maidenhead, UK

Ospina, S., Dodge, J., Foldy, E., Hofmann-Pimilla, A. (2008) Taking the Action Turn: Lessons from Bringing Participation to Qualitative Research. In Reason, P., Bradbury, H. (eds), The Sage Handbook of Action Research: Participative Inquiry in Action. Sage, London, UK

Reynolds, L., Merritt, R. (2010) Scoping. In French, J., Blair-Stevens, C., McVey, D., Merritt, R. (eds), Social Marketing in Public Health. OUP, Oxford, UK

Secure System? Challenge Accepted: Finding and Resolving Security Failures Using Security Premortems

Shamal Faily
Department of Computer Science
University of Oxford
shamal.faily@cs.ox.ac.uk

Simon Parkin
School of Computer Science
Newcastle University
s.e.parkin@ncl.ac.uk

John Lyle
Department of Computer Science
University of Oxford
john.lyle@cs.ox.ac.uk

Risk-driven approaches are dominant in secure systems design; these aim to elicit and treat vulnerabilities and the threats exploiting them. Such approaches, however, are so focused on driving risks out of system design, they fail to recognise the usefulness of failure as a vehicle for security innovation. To explore the role of failure as a design tool, we present the security premortem: a participative design technique where participants assume that a system has been exploited, and plausible reasons are given for explaining why. We describe this approach and illustrate how software tools can be used to support it.

Risk,Premortem,CAIRIS

1. INTRODUCTION

Many of the approaches associated with secure system design are driven by the elicitation and mitigation of risks. These are concerned with identifying vulnerabilities which expose *assets* of value, together with threats which exploit them. While useful concepts for both design and information security management, focusing too much on risks may draw undue attention to the symptoms of security failures, rather than their root causes. To better understand these causes we must elevate security failures to concepts worthy of analysis in their own right.

Because there are many reasons for why a system might be exploited, it has been argued that security is what social planners call a *wicked problem*. This is because we lack clarity about what it means to secure systems, tests for proving a system is secure, and a grasp of all possible solutions for satisfying a specified security problem (Faily and Fléchais 2010b). Therefore, while assurances may be given that a system's specification is secure, we can never be certain that circumstances won't arise where these assurances fail to hold. What specifying a design does do is force designers to make value judgements about what might be a *good enough* solution. Even if these judgements lead to ineffective design decisions, knowledge about failures still provide insights to designers about the nature of the problem space.

It seems nonsensical that we might want to make design decisions knowing that they are doomed to fail, but doing so is also emancipatory. While it is generally accepted that security is a *weak link* problem in that attackers will find and exploit this weak link, reflecting on the different ways a chain *might* break can lead to insights that would otherwise be missed if the weak link is allowed to fail and then quickly replaced with another functional — but still imperfect — link. The idea of thinking about the potentially broken chain rather than its weak link is analogous to the business scenario planning metaphor of the *premortem*. These operate on the assumption that a solution has failed and, rather than reflecting on what might have gone wrong, designers instead generate plausible reasons for explaining the solution's failure (Klein 2007). Even when ambiguity shrouds the reason for this failure, the lack of clarity provides clues about what additional evidence is needed before the "cause of death" can be established.

In this paper we present an approach for planning, running, and evaluating the results of a *security premortem*: a participative design technique where participants assume that a system has been exploited, and plausible reasons are given for

Proceedings of BCS HCI 2012 Workshops
Designing Interactive Secure Systems

explaining why. The objective of security premortems is to identify the reasons for a failure, rather than attempting to mitigate them. In Section 2, we present our approach, before illustrating how the software tools can support this technique, and integrate its results into the broader secure system design process in Section 3. Finally, in Section 4, we reflect on some of the consequences that might arise from running and evaluating premortems.

2. APPROACH

Our approach for running security premortems is loosely based on the three-step process proposed by Klein (Klein 2007); this is described in more detail in the following sections.

2.1. Presenting the scenario

In the first step the project team is brought together and informed that the project has failed because of security problems. Careful thought needs to be given to the "breaking-news" scenario being presented; it must be significant enough to cause the project's failure, believable enough for participants to take the failure seriously, yet also imprecise enough to yield several causes of failure. Based on an imaginary software platform we shall call ACME, an example of a possible scenario is described below:

A major news provider picked up a story based on blog reports by angry mobile phone users; these complain about increased email spam and phishing mails since they started using ACME services. This spam is sufficiently targeted that it evades spam filters. These incidents led to irate twitter posts appearing on the twitter feed on the ACME home page, especially from developers who users blame for this problem. As the bad press grew, major partners began to leave the project, and funding was cut. The cuts meant that the project was forced to stop work.

2.2. Stating potential causes of death

In the second step, team members are given time to independently write down every reason they can think of for the failure; this includes reasons they would normally consider inappropriate. Following this, the facilitator asks each person in turn for a reason, starting with the team leader or most senior team member present. These reasons may correspond to problems at different levels of abstraction. For example, one possible reason might be: *Hardcoded administrator accounts and secrets were, as a result of testing, committed in a major release of ACME that is used in most installations of ACME. This allowed attackers to target cloud services hosting ACME services, and leaking personal information to pastebin.*

2.3. Incorporating reasons into the design

After each reason has been recorded, participants review a hard-copy collection of project specification, reports, and models. Where these artefacts correspond with a possible reason, these are tagged by affixing a "reason" post-it note to the appropriate location in the physical document. Where an artefact does not but should, ideally, exist, then these are noted on a white-board or flip-chart and post-it notes are attached. At the end of the workshop, the team leader reviews the reasons; the tags are used to determine how these reasons cross-cut the system design. Based on this, an action plan is proposed for addressing these reasons.

3. SUPPORTING TOOLS

We now illustrate how software tools can be re-used or extended to support security premortems. In particular, we consider the open-source CAIRIS (Computer Aided Integration of Requirements and Information Security): a Requirements Management tool for supporting the elicitation and specification of usable and secure systems. CAIRIS was developed to implement the IRIS (Integrating Requirements and Information Security) meta-model, which integrates concepts from HCI, Requirements Engineering, and Information Security (Faily and Fléchais 2010a). CAIRIS was designed to be a research tool and can be extended to support new design concepts and techniques. A more detailed overview of CAIRIS is beyond the scope of this paper, but more information about its design and evolution can be found in (Faily and Fléchais 2012).

3.1. Presenting the scenario

Software tools like CAIRIS can be used to support the elicitation and specification of scenarios used in premortem workshops. These scenarios might be consequences of *misusability cases*: scenarios which describe how design decisions cause usability problems that might lead to system misuse (Faily and Fléchais 2011). These scenarios are motivated by argumentation models, the grounds of which might be requirements or architectural components which specify how the system should behave, or behavioural characteristics of personas — descriptions of archetypical user behaviour (Cooper 1999) — that use it. As such, these scenarios describe contexts where a system which satisfies the designers' intentions might be unintentionally exploited.

Scenarios might also arise from specified *misuse cases*; these describe the consequences of specified risks being realised (Sindre and Opdahl 2005). For example, (Atzeni et al. 2011) describes how, with the aid of CAIRIS, open-source threat data from the OWASP (The OWASP Foundation 2011) project was used to create personas for attackers and elicit the attacks they might employ.

3.2. Stating potential causes of death

The causes of failure naturally fit with the requirements engineering concept of *domain properties*; descriptive statements about the problem world. In the case of premortem scenarios derived from misusability cases, CAIRIS can associate these reasons with the both the scenarios and its argumentation model. This is illustrated by the example in Figure 1, which forms the basis of the scenario presented in Section 2.

3.3. Incorporating reasons into the design

To support the integration of premortem reasons into the requirements and architectural design, we have extended CAIRIS in two ways.

First, we have extended CAIRIS to support the association of one or more *tags* with model concepts. In addition to providing a means for interrogating CAIRIS models based on these tags as a search criteria, several of CAIRIS' visual models have also been updated to support the annotation of tags to different model elements.

Second, if risks are not evident then these tagged artefacts can be analysed in more detail. To allow this, we have built qualitative data analysis capabilities into CAIRIS. This allows us to assign codes — words or phrases that assign a summative, essence-capturing attribute for a portion of language based data (Saldaña 2009) — directly to design artefacts stored in CAIRIS. For example, based on the reason we gave in Section 2.2, we might wish to better understand the factors that might lead to this reason; as Figure 2 shows, this might include coding persona descriptions. Relationships drawn between these thematic concepts might be used to motivate vulnerabilities, in the same way that argumentation models can motivate premortem scenarios.

4. CONCLUSION

This paper presented an approach for applying premortems for finding security failures in a secure system design. We have also shown how CAIRIS can support this technique by facilitating scenario generation, categorising models according to failure reason and, based on these reasons, analysing model data to find ways of addressing their root causes.

We are currently evaluating both this technique and CAIRIS' ability to support it as part of the *webinos* project. As part of this evaluation, we are exploring possible stimuli that might be used by participants for eliciting reasons. These stimuli includes adopting the perspective of an attacker with the aid of pre-developed attacker personas. We are also evaluating the physical settings where premortem processes can be run. By running premortem workshops, participants gain respect from their colleagues by suggesting insightful reasons, and healthy team dissent is encouraged rather than discouraged (Klein 2009). However, workshops can be difficult to set up when team members are distributed, and a successful outcome is often dependent on the effectiveness of the group's facilitator. For this reason, we are currently investigating how effective the premortem process might be if reasons are elicited on a one-to-one basis outside of a workshop, and what sort of factors might lead to the elicitation of insightful reasons given the change of setting.

By tool-supporting premortems, we also raise the question of how far tools can go before they obstruct, rather than stimulate, creativity and innovation? The *innovation design dilemma* suggests that structure might stymie creativity but, without it, creative output might become too disruptive (Hobek 1988). In this respect, we believe CAIRIS strikes a balance. By providing only modest tool-support during workshop settings, the tool provides little to obstruct group dynamics. Also, by aligning reasons and their rationale with CAIRIS models, the impact of security innovation arising from premortem scenarios can be explored.

If qualitative data analysis is to be used to find the root causes of failure then CAIRIS will need to be used more visibly in group settings; this will help mediate discussions around qualitative models stored within the tool. While techniques for using software for supporting qualitative *research* are well known, their use for supporting *design*, especially for security, is ill-explored. Consequently, future work will also explore the effectiveness of qualitative data analysis techniques in conjunction with premortems to more directly support secure system design activities.

5. ACKNOWLEDGEMENTS

The research described in this paper was funded by the EU FP7 *webinos* project (FP7-ICT-2009-05 Objective 1.2).

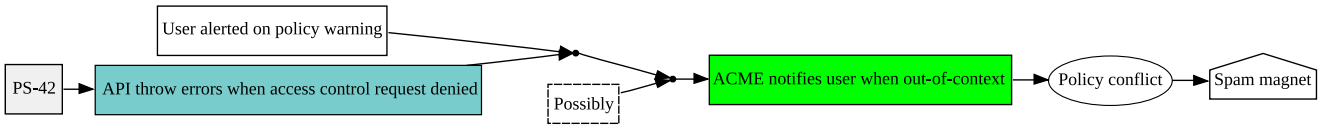

Figure 1: *Misusability case argumentation model motivating a security premortem*

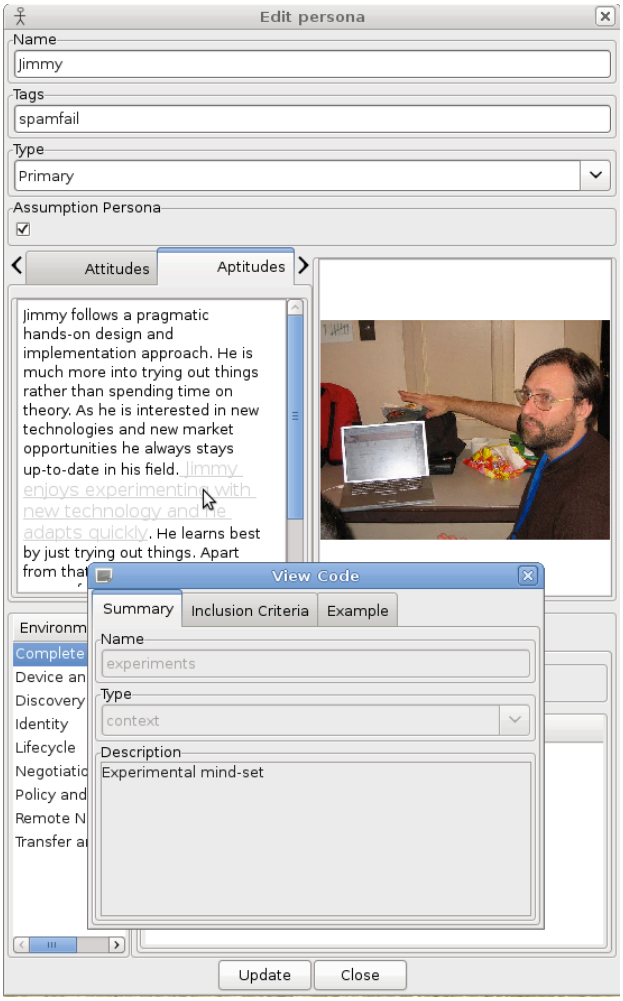

Figure 2: *Coding a persona based on a premortem reason*

REFERENCES

Atzeni, A., Cameroni, C., Faily, S., Lyle, J., and Fléchais, I. (2011). Here's Johnny: a Methodology for Developing Attacker Personas. In *Proceedings of the 6th International Conference on Availability, Reliability and Security*, pages 722–727.

Cooper, A. (1999). *The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity (2nd Edition)*. Pearson Higher Education.

Faily, S. and Fléchais, I. (2010a). A Meta-Model for Usable Secure Requirements Engineering. In *Proceedings of the 6th International Workshop on Software Engineering for Secure Systems*, pages 126–135. IEEE Computer Society.

Faily, S. and Fléchais, I. (2010b). To boldly go where invention isn't secure: applying Security Entrepreneurship to secure systems design. In *Proceedings of the 2010 New Security Paradigms Workshop*, pages 73–84. ACM.

Faily, S. and Fléchais, I. (2011). Eliciting Usable Security Requirements with Misusability Cases. In *Proceedings of the 19th IEEE International Requirements Engineering Conference*, pages 339–340. IEEE Computer Society.

Faily, S. and Fléchais, I. (2012). Software for Interactive Secure System Design: Lessons Learned Developing and Applying CAIRIS. In *Designing Interactive Secure Systems: Workshop at British HCI 2012*. To Appear.

Hobek, J. (1988). The innovation design dilemma: some notes on its relevance and solution. In Grønhaug, K. and Kaufmann, G., editors, *Innovation: a cross-disciplinary perspective*. Norwegian University Press.

Klein, G. (2007). Performing a project premortem. *Harvard Business Review*, 85(9):18–19.

Klein, G. A. (2009). *Streetlights and shadows: searching for the keys to adaptive decision making*. MIT Press.

Saldaña, J. (2009). *The coding manual for qualitative researchers*. Sage.

Sindre, G. and Opdahl, A. L. (2005). Eliciting security requirements with misuse cases. *Requirements Engineering*, 10(1):34–44.

The OWASP Foundation (2011). Open Web Application Project (OWASP) web site. `http://www.owasp.org`.

Storytelling for Tackling Organized Cybercrime

Muhammad Adnan Tariq
Royal Institute of Technology
SE-100 44 Stockholm, Sweden
tari@kth.se

Joel Brynielsson
Royal Institute of Technology
SE-100 44 Stockholm, Sweden
joel@kth.se

Henrik Artman
Royal Institute of Technology
SE-100 44 Stockholm, Sweden
artman@kth.se

Cybercrime is often organized, and the actual individuals that are perpetrating the different parts of the attack might not be aware of or interested in the overall organizational motives behind the attack. In this paper, based on interviews with IT security experts, we build on the attacker persona methodology and extend it with methodology to also handle organizational attacking motives in order to tackle organized cybercrime. The resulting framework extends the attacker persona methodology by also using narratives in order to assess the own organization's security. These narratives give rise to intrigue sketches involving any number of attacker personas which, hence, make it possible to take organized cybercrime into account.

Organized cybercrime, narrative, attacker persona, intrigue sketch

1. INTRODUCTION

From a user perspective, the problem of not being able to effectively apply security mechanisms is twofold: lack of usability in the security mechanism itself (Zurko and Simon 1996; Zurko 2005) and lack of user engagement due to not understanding the implications of bypassing a security mechanism (Platt 2006). As an example, Whitten and Tygar (1999) highlighted how users were unable to understand the security mechanism (PGP 5.0) which eventually lead to confidential data being sent in the clear. Similarly, users are in most cases not well aware about the consequences of their actions which can lead to devastating results (Adams and Sasse 1999; Fléchais and Sasse 2009).

Consequently, there is a need for a framework to be used for enlightening the user/defender about the attacker perspective (Brynielsson 2009), and enable them to specify security-centric requirements in their context of use. However, in order to do this one must have some representation of the threats and the actual actors who might pose the threat. Still, such criminal actors are likely to be hard to find and even harder to interview. In this paper we follow-up on recent work (Atzeni et al. 2011) and propose a solution based on the persona methodology.

2. ORGANIZATIONAL SECURITY ASSESSMENT

The elastic nature of the general and routine-like use of the term user as identified by Cooper (2004) is being acknowledged by many researchers and forms the basis for the use of personas in systems development. However, we argue that problems, and explicitly security problems, can be as elastic, especially in terms of assessing the organizational security. As an example, consider a situation where a user somehow downloaded a malicious file from the Internet. This whole activity points towards multiple factors which could have resulted in the download of that file. Such factors typically represent inadequacies with regard to, e.g., the security policy, the security mechanism, the user awareness, and so forth. The security problem in itself is elastic and depends not only on a single factor, but rather upon multiple factors. In this paper, a *narrative* is taken to be an indicator pointing towards such factors.

To further elaborate on the narrative property, consider the known analogy of the elephant and the six blind men. The blind men come across an elephant; by feeling different parts of the elephant each individual tries to describe what they perceive: they will all describe the elephant in various, and probably different, ways depending on if they have encountered the tail, the ears, the legs, the proboscis, or any other part of the elephant. This situation highlights that any complex and large problem being immediately perceived by an individual may elicit many different descriptions. In terms of an organization, the elephant represents the security-critical issues/problems, e.g., the download of a malicious file, and the blind men denote the different stakeholders. The perceptions of these stakeholders are the narratives, and each stakeholder might be able to describe an event or

Proceedings of BCS HCI 2012 Workshops
Designing Interactive Secure Systems

activity using a number of narratives. The narrative provides us with potential causes of an event, and with multiple people providing their narratives it becomes easier to identify overall security holes. Of course, the most predominant cause of the security issue will have an overlapping effect among the collected narratives. This overlapping between narratives will identify the major loop holes, and the collection of narratives will incorporate factors which one individual was unable to identify. Thus, the collection of narratives encompasses multiple factors and provides insight into the cause of the security problem from different angles.

2.1. Organized cybercrime and personas

Recent trends in the IT security landscape suggest that organized cybercrime has become part of the everyday cyber landscape with criminal groups using cybercrime to achieve their goals (McCusker 2006; Choo and Smith 2008). Moreover, McCombie and Pieprzyk (2010) suggest that there are cases where groups of cybercriminals have used extortion, blackmailing, and online fraud to achieve their desired goal. To map such an organization into a persona is a challenge due to the inadequacy of observable data about organizational culture, environment, hierarchical structure, communication, etc. Furthermore, the persona methodology is designed towards convergence of a group of individuals with more or less similar motivations, goals, skills, behavior, etc., into a single personification. To overcome these issues, the persona methodology needs to be extended to provide insight into such critical issues. However, there has been work carried out to capture the group or organizational aspect of persona (Giboin 2011; Judge et al. 2012; Matthews et al. 2011), but personification of a group of attackers has its limitation mainly due to the secret nature of such organizations.

3. FRAMEWORK

In this section we present our framework, which is an attempt to highlight the organizational security threats while extending the persona methodology. The framework comprises 1) narratives, 2) attacker personas (including scenarios), 3) intrigue sketches, and 4) plots. Narratives have already been discussed while this section serves to describe attacker personas, intrigue sketches and plots.

3.1. Attacker personas

Personas is a method for highlighting end users and their needs of a system (Cooper 2004). Since personas can be used to replace direct user participation its usefulness has been questioned by some people (Grudin 2006; Portigal 2008). However, others argue that this is its actual strength since actual user involvement in the design can be perceived as a hinder due to idiosyncratic demands of the real users (Cooper 2004; Grudin 2006). By representing the attackers as personas we can get an understanding of the complex ways attackers might work. This introduces problems as we cannot interview actual attackers. Atzeni et al. (2011) have dealt with this problem by using assumptions of their character while collecting data from sources such as attacker taxonomies, profiling, and knowledge elicitation workshops. However, Tariq et al. (2012) argue that the personas should be context independent as security is not a single context problem: in fact, each security issue has multiple contexts, especially in terms of organizations. De-attaching the context from the attacker personas gives the flexibility to use attacker personas in multiple contexts. That is, we do not argue against a context bound framework but argue against an attacker persona that is bound to specific contexts or specific systems. Rather, we perceive attacker personas as a collection of threats to an organization.

Scenarios are part of the persona methodology and are used to describe the sequential activities that a user undertakes to reach a specific goal. We have used the concept of scenarios, as discussed by Quesenbery (2006), and applied it in terms of attacker activities, i.e., we have developed a set of small stories which emphasize how a specific attacker in the past has attacked several organizations to achieve their goal. However, these stories do not provide a detailed step by step approach to describe an attack, but rather provides a high-level description of the attack. The aim of using the scenarios is to provide a basic understanding of how an actual attacker could operate and which weaknesses that might be exploited by the attacker. This information is particularly helpful while analyzing the narratives and relating them with the attacker personas. Hence, the idea of presenting this information is to provide a guideline so that the narrative can be related to the personas and scenarios while developing *intrigue sketches*, which will be discussed further in the following section.

3.2. Intrigue sketches

Before we define the intrigue sketch it is necessary to understand why we need intrigue sketches. As discussed in Section 3.1, our personas are context independent so in order to put them in an organizational context we need to relate them to organizational-specific narratives. In practice, this process consists of a systematic interpretation of the narrative in terms of attacker personas. The interpretation can mainly be carried out by someone

who has a good understanding of IT security, and thus the security analyst is part of the process. This interpretation of a narrative in terms of personas enables one to understand the problem identified by the narrative from an IT security viewpoint. Also, taking this attacker perspective could help determining the overall motivations and goals behind an attack, which might lead to identifying organized cybercrime activity by looking at multiple intrigue sketches, which will be discussed further below.

The intrigue sketches make use of narratives, security analysts and attacker personas with scenarios. Both the narrative and the attacker personas have some attributes in common which are mainly goals, motivations, and skills. The narrative incorporates these aspects from the respondent perspective, e.g., how a certain event took place, which critical asset was targeted, and so forth. Similarly, each persona contains a set of goals, motivations, and skills. When these attributes, derived from a narrative and the corresponding attacker personas, are related with each other by a security analyst/expert the result is an intrigue sketch. The intrigue sketch holds information about the relevant attacker or attackers, possible attack procedure (derived from the corresponding attacker persona scenario), motivations, and goals. The intrigue sketch development process can be seen as a way to combine the attacker perspective (personas with scenarios), the respondent perspective (narrative) and the security perspective (the security analyst) in order to understanding the multidimensional aspects of security. For the development of the overall framework, it should also be emphasized that each intrigue sketch will contain at least one persona, but can of course contain more depending on the narrative. To make sense of the intrigue sketches in terms of the organizational perspective, each intrigue sketch should be classified mainly on the basis of the attacker's goals and in some cases the combination of both goals and motivations. This classification of the intrigue sketches will prove necessary in the next phase of the framework, which is the creation of *plots*.

3.3. Plots

The plot is the last part of the framework, which describes the overall security of the organization by relating intrigue sketches with the existing security practices being used by the organization. Each intrigue sketch can be related with the existing security practices of the organization either individually or collectively to point out threats to the organization. However, using intrigue sketches individually may result in ignoring the multidimensional aspect of security. On the other hand, however, there could be a case where the intrigue sketch represents an isolated attacker's

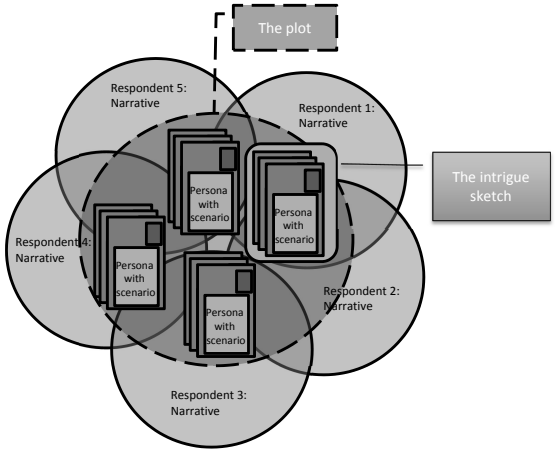

Figure 1: *The complete framework consists of different narratives that are collected from the respondents in the organization, which are then being related with attacker personas with scenarios in order to develop intrigue sketches, which are finally brought together with existing organizational practices to develop the overall plot.*

activities. In such case, the plot will comprise of a single intrigue sketch related with the organizational practices to identify potential threats. A collective usage of the intrigue sketches will provide a holistic view of the organizational security. To achieve this it is critical that the intrigue sketches are specified so that it is easy to identify the overlapping among them. This problem is solved by the specification of intrigue sketches in terms of goals and motivations, as mentioned earlier. The intrigue sketches can be related by using a combination of both goals and motivations, e.g., attackers who are trying to steal critical information and are ideologically motivated can be clustered together, etc.

Once the intrigue sketches have been synthesized they can be related to existing organizational practices, which will result in an assessment of the existing security practices of the organization and eventually identify threats that the organization might face. However, it should be mentioned that the number of plots will depend upon the number of intrigue sketch syntheses, i.e., the intrigue sketches might result in one espionage synthesis and one mafia synthesis which, when related with the organizational practices, will yield two different plots since they represent two separate kinds of attacks. To finally tackle the organized cybercrime threat, the attacker personas can be related from an organized cybercrime perspective based on their goals and motivations to find out whether the attacker personas represent individual attackers or are part of an organized criminal activity. To summarize, see Figure 1 where the framework constituents have been put in perspective to each other.

4. CONCLUSIONS

We have presented a framework to be used for understanding the IT security environment in an organization. The framework highlights possible inconsistency in terms of understanding the requirements and expectations from an organizational perspective. Also, the framework is an effort to assess the organizational security from multiple perspectives by extending the persona methodology. We have presented attacker personas such that they are context independent and are used to incorporate the organized cybercrime perspective. The major contribution is the intrigue sketch which is the combination of a respondent's narrative, generic attacker personas and a security specialist's assessment. The intrigue sketch sets a scene for the possibility to frame one or several attackers in a specific situation.

REFERENCES

Adams, A. and M. A. Sasse (1999, December). Users are not the enemy. *Communications of the ACM*, 42(12), pp. 40–46.

Atzeni, A., C. Cameroni, S. Faily, J. Lyle, and I. Fléchais (2011, August). Here's Johnny: a methodology for developing attacker personas. In *Proceedings of the Sixth International Conference on Availability, Reliability and Security (ARES 2011)*, Vienna, Austria, pp. 722–727.

Brynielsson, J. (2009, March). An information assurance curriculum for commanding officers using hands-on experiments. *ACM SIGCSE Bulletin*, 41(1), pp. 236–240.

Choo, K.-K. R. and R. G. Smith (2008, June). Criminal exploitation of online systems by organised crime groups. *Asian Journal of Criminology*, 3(1), pp. 37–59.

Cooper, A. (2004). *The Inmates Are Running the Asylum: Why High-Tech Products Drive Us Crazy and How to Restore the Sanity* (2 ed.). Indianapolis, IN: Sams Publishing.

Fléchais, I. and M. A. Sasse (2009, April). Stakeholder involvement, motivation, responsibility, communication: How to design usable security in e-Science. *International Journal of Human-Computer Studies*, 67(4), pp. 281–296.

Giboin, A. (2011, February). From individual personas to collective personas. In *Proceedings of the Fourth International Conference on Advances in Computer-Human Interactions (ACHI 2011)*, Guadeloupe, France, pp. 132–135.

Grudin, J. (2006). Why personas work: The psychological evidence. In J. Pruitt and T. Adlin (Eds.), *The Persona Lifecycle: Keeping People in Mind Throughout Product Design*, San Francisco, CA: Morgan Kaufmann. Chapter 12, pp. 642–663.

Judge, T., T. Matthews, and S. Whittaker (2012, May). Comparing collaboration and individual personas for the design and evaluation of collaboration software. In *Proceedings of the 30th Conference on Human Factors in Computing Systems (CHI 2012)*, Austin, TX, pp. 1997–2000.

Matthews, T., S. Whittaker, T. Moran, and S. Yuen (2011, May). Collaboration personas: A new approach to designing workplace collaboration tools. In *Proceedings of the 29th Conference on Human Factors in Computing Systems (CHI 2011)*, Vancouver, Canada, pp. 2247–2256.

McCombie, S. and J. Pieprzyk (2010, July). Winning the phishing war: A strategy for Australia. In *Proceedings of the Second Cybercrime and Trustworthy Computing Workshop (CTC 2010)*, Ballarat, Australia, pp. 79–86.

McCusker, R. (2006, December). Transnational organised cyber crime: distinguishing threat from reality. *Crime, Law and Social Change*, 46(4–5), pp. 257–273.

Platt, D. S. (2006). *Why Software Sucks... and what you can do about it*. Boston, MA: Addison-Wesley.

Portigal, S. (2008, January–February). True tales: Persona non grata. *interactions*, 15(1), pp. 72–73.

Quesenbery, W. (2006). Storytelling and narrative. In J. Pruitt and T. Adlin (Eds.), *The Persona Lifecycle: Keeping People in Mind Throughout Product Design*, San Francisco, CA: Morgan Kaufmann. Chapter 9, pp. 520–554.

Tariq, M. A., J. Brynielsson, and H. Artman (2012, August). Framing the attacker in organized cybercrime. In *Proceedings of the IEEE European Intelligence and Security Informatics Conference 2012 (EISIC 2012)*, Odense, Denmark.

Whitten, A. and J. D. Tygar (1999, August). Why Johnny can't encrypt: A usability evaluation of PGP 5.0. In *Proceedings of the 8th USENIX Security Symposium*, Washington, DC, pp. 169–183.

Zurko, M. E. (2005, December). User-centered security: Stepping up to the grand challenge. In *Proceedings of the 21st Annual Computer Security Applications Conference (ACSAC 2005)*, Tucson, AZ, pp. 187–200.

Zurko, M. E. and R. T. Simon (1996, September). User-centered security. In *Proceedings of the 1996 New Security Paradigms Workshop (NSPW'96)*, Lake Arrowhead, CA, pp. 27–33.

Designing acceptable user registration processes for e-services

Chris Porter
Dept. of Computer Science
University College London
London, United Kingdom
c.porter@cs.ucl.ac.uk

M. Angela Sasse
Dept. of Computer Science
University College London
London, United Kingdom
a.sasse@cs.ucl.ac.uk

Emmanuel Letier
Dept. of Computer Science
University College London
London, United Kingdom
e.letier@cs.ucl.ac.uk

User registration can have a serious impact on the success of online government services. Different services require different levels of identity assurance, and different registration processes are put in place to deliver them. But from the citizen's perspective, these processes often require a disproportionate amount of effort, which reduces users' acceptance. Typically, when sign-up to high-effort services is not mandatory, take-up is low; when it is compulsory, it causes resentment, and neither is desirable. Designers of services requiring registration currently have no way of assessing likely user acceptance at design time. We are introducing a tool that allows system designers to identify the impact of registration processes on different groups of users, in terms of workload and friction. Personas have been successfully applied to assist security designers, and we extend the concept with statistical properties, and introduce the Persona Group Calibration (PGC) exercise to calibrate the different personas for sensitivity to specific identity-related elements.

Registration, E-services, Security friction, Workload, Design

1. INTRODUCTION

The registration process for any e-service can have a dramatic impact on the user's lived experience. There is empirical evidence that cumbersome registration processes reduce the number of service users (egovbarriers.org 2007). Security measures adopted should not be a burden on users (OECD 2007). User behaviour is goal-driven – they sign up to an e-government service because they need to complete a task (Sasse & Flechais 2005), and barriers to completing such tasks have a significant negative impact on the user's lived experience (Inglesant & Sasse 2007). If registration processes for e-services are too cumbersome, citizens are discouraged in the earliest stages, and may never experience the potential benefits of transacting online. To support designers of e-services, we have developed a citizen-centric technique that allows us to capture the user's sensitivity to registration process design elements and help us predict the expected workload and level of friction (chances that a user leaves the process) caused by new registration processes. The technique is based on an empirical exercise (Persona Group Calibration) to identify causes of friction within the registration process. This information is then used to predict expected reactions to new designs across different projects. In the Compliance Budget, Beautement, Sasse & Wonham (2008) define friction as the imbalance between the business process (user goals) and security behaviour required, including any inherent cognitive and physical workload. In our model, workload is measured as a separate factor since it was found that workload and friction are not always linearly related: registration pages with high workload values might still result in low friction, for instance when online access makes their lives easier (e.g. because it removes the need to travel to an office only open for certain hours). Thus we introduce the Type of Service (i.e. level of regular compulsion) to explain this phenomenon.

2. THE METHOD

The following section describes the method we have developed to capture user's sensitivity to friction in e-service registration processes, followed by an outline of the process to apply this information in a prediction model to forecast friction (including workload).

2.1 Setting the foundations

In this section we will establish the set of design elements that have a negative impact on the user's lived experience (causing a negative reaction, such as frustration and even service abandonment). We

conducted an empirical study to determine the main points that cause workload and frustration in e-service registration processes. Five design elements were identified. These are grounded in empirical data and discovered through the adoption of techniques borrowed from *Grounded Theory*: *open* and *axial coding* (Charmaz 2006). These design elements are summarized below:

- **Items to recall (ItR)**: Number of facts the user has to recall from memory (e.g.SSN)
- **Items to generate (IG)**: Number of new credentials required (e.g. username/password)
- **Delay (D)**: An indicator denoting whether the production task is delayed by a security task (e.g. waiting for an activation email – minor delay – or waiting for the provider to manually validate a registration form – major delay)
- **Interruption of routine (I)**: A flag indicating whether the user has to go out of his way to complete the task (e.g. visit a registration authority)
- **Perceived workload (W)**: The perceived level of cognitive and physical workload induced by the security tasks

We also noted that these design elements are weighted differently depending on the regularity of compulsion for an e-service being discussed. For this purpose, we had to consider the **Type of Service (ToS)** as a behavioral modifier**.** The Type of Service can be defined as: *the number of times users are legally obliged to use a service in any given year*. These can be summed up in 4 levels:

- **Level 1**: No legal obligation to use the service
- **Level 2**: Legal obligation to use service at most a couple of times in a lifetime
- **Level 3**: Legal obligation to use service at most once per year
- **Level 4**: Legal obligation to use service multiple times per year

At Levels 3 and 4 penalties apply when citizens do not comply with set regulations (e.g. deadlines), while benefits exist for compliance. For Levels 2, 3 and 4 citizens can alternatively send forms by post.

2.2 Persona Group Calibration (PGC)

Adhering to citizen-centric design principles we developed personas following Cooper's recommendations and through successive refinements, starting from a plausible approximation of our user, supplementing it with experience, interviewing and secondary data, we move towards a fictitious user archetype having specific characteristics, needs and goals (Cooper 2004). To predict the expected reactions of different personas towards specific design elements in registration processes, we first need to understand how persona representatives behave when facing different tasks. These representatives participate in a Persona Group Calibration (PGC) exercise which in turn provides us with a set of behavioural parameters. Representatives of a persona under investigation may obtain similar results to representatives of another persona. In this case, these two personas can then be grouped under a single persona group. A persona group encompasses one or more personas that share a common factor: *behaviour when facing different design elements in registration processes.*

We needed to set the assessment in the context of a set of pre-defined registration tasks; to identify these, we surveyed the registration processes on a number of e-government portals, and for each eService identified, we measured design elements defined in Section 2.1, except for workload. Perceived workload is user-specific, and can only be measured during calibration. From this exercise the most common registration page setups used in e-services were generalized into nine different fictitious registration processes. The registration processes cover as many configurations as possible in order to capture the widest range of data from test participants. Extreme configurations are also present within the set of nine processes (e.g. from a simple email/password registration process and up to lengthy and laborious processes which also require physical travelling).

Based on the pre-defined tasks, we built a mechanism that helps us capture user behaviour, providing us with enough data to be able to predict friction on different configurations of design elements. We created a fictitious government portal offering 9 e-services with different registration processes. PGC participants were asked to go through each registration process. After each task, the participant was asked to rate 6 workload scales assessing the different dimensions as specified in NASA-TLX (Mental Demand, Physical Demand, Temporal Demand, Performance, Effort and Frustration). Following the 9 tasks, the participant was asked to give a weighting for each of the six scales by completing a pair-wise comparison exercise. For a full discussion on NASA-TLX, the reader is directed to Hart & Staveland (1988). After each of the nine registration tasks, the participant is also required to state whether he/she would consider completing the process (given the current registration process configuration), and such decision needs to be taken in four situations, one for each level defined in the *Type of Service* design element. To rate friction, participants are asked the following question: *"Given this registration process, would you consider registering for this service?"* Four 10-point Likert scales ranging from 0 to 1 (with

increments of 0.1) are presented, one for each of the four *Type of Service* levels.

After completing the 9 tasks, the participant's data collected was transferred to a spreadsheet for further processing. Each sheet contains 9 rows, (one for each task) whereby each row holds the task ID, rating for each NASA-TLX workload scale, a computed overall weighted workload measure for each task, and friction for the four *type of service* levels. Once the participants from a specific persona group complete the PGC exercise, all of their data is merged and prepared for further processing. To be able to predict friction and workload we first need to fit two regression models on our data: a) *Linear Regression Model* for workload and b) *Binary Logistic Regression Model* for friction. After fitting these two models on the data (using a statistical package), we are provided with a *y* intercept (b_0) together with a set of regression coefficients, one for each design element, explaining the model's fit on our data. These coefficients can be defined as a persona group's behavioural properties with regards to the different design elements present in registration processes (see Section 2.1). These coefficients are then associated with the persona group under investigation. We found that certain design elements (predictors) are not statistically significant in the prediction of workload and friction. For this purpose and following proper model fitting techniques and statistical tests, only the most significant elements are used for both friction and workload models. Friction was best explained by *IG, D, I* and *Type of Service*, while *Workload* was best explained by *ItR, D* and *I*. This process allows us to capture a specific persona group's sensitivity to different design elements (using regression coefficients). We now apply these insights to elicit the level of friction and workload in new (or existing) registration processes. For this purpose one final step is required. For a specific registration process, we need to determine the values for each of the design elements defined in section 2.1. These are the required predictors, which together with the regression coefficients determined in the previous step, would allow us to generate friction and workload values. The regression coefficients, *y* intercepts and predictor values are then parameterized into the following two equations:

$$Y_i = b_0 + b_1X_{1i} + b_2X_{2i} + b_3X_{3i} + \ ... + b_nX_{ni}$$

The first equation is the linear regression model, where Y is the outcome variable (predicted worklaod).

$$P(Y) = \frac{1}{1 + e^{-(b_0+b_1X_{1i}+b_2X_{2i}+b_3X_{3i}+\cdots+b_nX_{ni})}}$$

The second equation is a binary logistic regression model, where P(Y) is the probability that a person completes the registration process, where 1-P(Y) is defined as the probability that a person abandons the registration process.

In both cases, b_0 is the *y* intercept for the model while b_n is the coefficient for the corresponding predictor X_n (design element value). These coefficients will vary across different persona groups based on the output from the PGC exercise.

After calculating Y and P(Y), we obtain a grounded idea of how a particular persona group reacts to a given design. Given the designers objectives (e.g. friction < 10%) an iterative process commences whereby the registration process is revisited, modified and re-assessed as part of a larger business process. This helps designers achieve a balance between the level of identity-assurance required (security goals) and the friction caused on the business process from the perspective of different user groups.

3. CASE TOOL

The method described in Section 2 is laborious, therefore we developed a decision support system to assist decision makers in the iterative assessment of design alternatives. Persona groups are stored in a persona library, making them available for re-use in different projects. This collaborative web-portal was developed using ASP.Net MVC 3. SPSS is used to generate the statistical parameters after each PGC exercise.

4. CASE STUDY

Formative evaluation of the method and corresponding tool was carried out through a real-world case study. A collaborative agreement was set up with a government agency in Malta (Employment and Training Corporation - ETC).

4.1 Objectives

ETC's management wanted to create an e-service to be used by the majority of human resource managers on the island. This requires a registration process that is acceptable by, and that does not add considerable burdens on users.

4.2 Method

We considered the HR Manager persona group for this study, and a number of representatives from several leading IT firms were contacted to participate in the first PGC exercise. Data was collected and prepared for processing. The two regression models were applied and the respective

sets of coefficients (and *y* intercepts) were generated and assigned to the persona group under investigation. This allowed us to analyse the impact that the registration process might have on this persona group. The registration process required users to visit a registration authority in person and after verifying their identity, a PIN would be sent by post. Once received, a new password is requested in order to activate the account. This can be annotated as follows:

Table 1: *Annotation of proposed registration process*

Security Element	Measurement	Details
IG	2	*Password and PIN*
ItR	2	*ID No., and email address*
D	Major	*Wait for PIN by post*
I	True	*Visti a registration authority*
ToS	3	

4.2 Results

The predictors (design element measurements) were parameterized into the regression equations, together with the respective *y* intercepts and regression coefficients, and values for friction and workload were obtained. Projected friction given the registration process and type of service under consideration was of 44% (i.e. 1-P(Y) == (1-0.557)*100)) with a workload of 71%. This meant that almost half of the potential users would have abandoned the registration process and opted for alternative means. This has led management to reconsider their original plans and revert to alternative registration processes. One option was to offset the physical workload by requesting additional information, verifying such data manually while eliminating the need to visit a registration authority. A new registration process was devised with the following measurements:

Table 2: *Annotation of redesigned registration process*

Security Element	Measurement	Details
IG	2	*Password and PIN*
ItR	17	*More identifying info required*
D	Major	*Manual verification by ETC staff*
I	False	*No interruptions on daily routines*
ToS	3	

This resulted in improved values with friction at just over 10% - however, workload increased to 75%. In situations where the *Type of Service* is high people are ready to accept higher levels of workload in order to *gain access* to this kind of service (making their lives easier for future interactions), hence the low level of friction.

5. CONCLUSIONS

Beautement, Sasse & Wonham (2008) presented the Compliance Budget paradigm which denotes that compliance issues are mainly caused by friction between the required security behaviour and the user's goals. Our work helps to quantitatively approximate the point at which users decide not to comply with the required security in registration pages. We plan to adopt and extend Faily & Fléchais' Persona Cases (Faily & Fléchais 2011). These personas, grounded in empirical data, would be associated with persona groups adding behavioural knowledge to such, which knowledge is in turn used to predict friction and inherent workload. Giving a 'voice' to personas through predictive statistical parameters, allows designers to make informed design decisions based on measurable and comparable values. Larger PGC exercises (with more participants) will result in more fine-tuned predictions however a statistical saturation point exists. The first case study gives a clear indication that the mechanics of the method (and corresponding tool) yield useful information. The captured knowledge on persona groups under investigation can be reused across different projects. We are confident that this method, and associated tool, will help designers garner further insights on their users which would in turn improve design decisions.

6. REFERENCES

Beautement, A, Sasse, A & Wonham, M 2008, 'The Compliance Budget: Managing Security Behaviour in Organisations', Proceedings of the 2008 workshop on New security paradigms (NSPW), 2008.

Charmaz, K 2006, Constructing Grounded Theory: A practical guide through qualitative analysis, SAGE Publications Ltd, London.

Cooper, A 2004, Inmates Are Running the Asylum, The: Why High-Tech Products Drive Us Crazy and How to Restore the Sanity, SAMS, Indiana.

egovbarriers.org 2007, Deliverable 3: Solutions for eGovernment - Section 3: Solutions for the seven eGovernment barriers, viewed May 2012, <http://www.egovbarriers.org/?view=project_outputs>.

Faily, S & Fléchais, I 2011, 'Persona Cases : A Technique for Grounding Personas', Proceedings of the 29th international conference on Human factors in computing systems, no. 2011, pp. 2267-2270.

Hart, SG & Staveland, LE 1988, 'Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research', Human Mental Workload, <http://humansystems.arc.nasa.gov/groups/TLX/tlxpublications.html>.

Inglesant, P & Sasse, AM 2007, 'Usability is the best policy: public policy and the lived experience of transport systems in London', People and Computers XXI - HCI , no. 21, pp. 35-44.

OECD 2007, 'Electronic Authentication and OECD Guidance for Electronic Authentication', OECD.

Sasse, AM & Flechais, I 2005, 'Usable Security: Why do we need it? How do we get it?', in Security and Usability, O'Reilly.

Towards the Systematic Development of Contextualized Security Interventions[1]

Steffen Bartsch
CASED – TU Darmstadt
Darmstadt, Germany
steffen.bartsch@cased.de

Melanie Volkamer
CASED – TU Darmstadt
Darmstadt, Germany
melanie.volkamer@cased.de

Current warnings during daily Web browsing demonstrate how difficult it is for developers to craft precise and comprehensible security interventions. While researchers have found that personal contextualization of interventions help in security-critical applications, taking this approach leads to an overwhelming range of options of how and when to intervene as well as which factors to consider. To make contextualized security interventions feasible, we need to support developers in selecting the relevant factors for their applications and support them in deriving the appropriate intervention strategy and content. In this paper, we propose a security intervention framework and methodology which provides such a support.

Security intervention, security communication, usability engineering

1. INTRODUCTION

Warnings of self-signed certificates in Web browsing are an example of how difficult it is to craft precise and comprehensible security interventions[2]. These warnings occur independently from the user's intention like browsing for information (a low risk) or transferring money (a high risk). This imprecision results in habituation that might cause users to ignore warnings in critical situations, since they have ignored them several times without any negative consequences. The problem here lies in the *precision* of this particular security intervention: The warning about the security of the connection should ideally only occur if there is a risk for the user from continuing to use the Web service as intended. Moreover, the warning is formulated in a technical language that is not comprehensible by the user: The inadequate *content* of the intervention prevents the user from understanding the risks of continuing to the Web site.

Several studies have shown that common Web browsing warnings (certificate warnings, Sunshine et al. (2009)) as well as other security indicators (passive indicators, Whalen and Inkpen (2005)) are not effective. The ineffectiveness is caused by the traditional approaches on security interventions that take the form of generic hazard warnings: warn a broad audience with static texts and symbols (Wogalter 2006). Accordingly, researchers propose to personalize and contextualize security indicators (De Keukelaere et al. 2009). The idea is to employ additional information on the context (e.g. user intention) and the user (e.g. expertise) so as to make better decisions on when to warn, how, and with what content.

However, this is challenging for developers of security-critical applications, because they need to take into consideration both the contextual factors and user characteristics in order to implement the correct *intervention strategy* (whether, when, and how to intervene) and *intervention content* (what content to convey). Developers need to evaluate available contextual factors, particularly for their availability and impact. They also have to combine the factors and balance whether the risks justifies a blocking warning or whether the negative effects (habituation, annoyance) are too high (cf. Böhme and Grossklags 2011). De Keukelaere et al. (2009) proposed an architecture for contextualized warnings that evaluates factors so as to decide which type of warning to display, but did not provide a methodology to select the relevant factors. Moreover, the content of the intervention should relate to the user to make the warning more convincing – for example, by taking the mental model into account (Bravo-Lillo et al. 2011).

[1]The work presented in this paper is supported by funds of the Federal Ministry of Food, Agriculture and Consumer Protection (BMELV) based on a decision of the Parliament of the Federal Republic of Germany via the Federal Office for Agriculture and Food (BLE) under the innovation support programme.

[2]We consider security interventions as signals to humans that influence security-relevant decisions, e.g. a green location bar (positive intervention) or warnings (negative intervention).

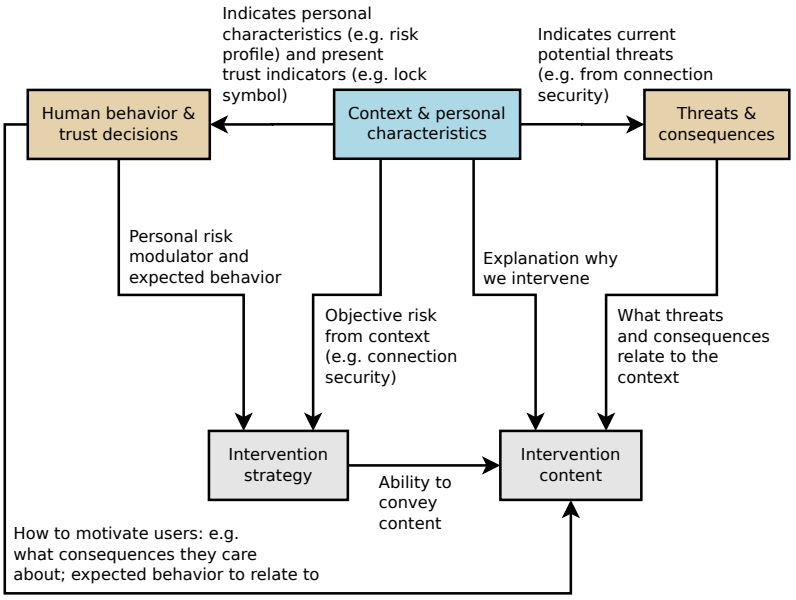

Figure 1: *Intervention framework with influences of the factors on intervention strategy and content*

One part of this challenge is to assess which threats and consequences are present in the actual situation, and whether to emphasize the technical threat (e.g. man-in-the-middle attack) or potential consequences (e.g. financial losses).

Developers need to combine different kinds of interventions (e.g. passive symbols, active warnings) for optimal results so that developers need broader support that considers a range of intervention options. However, prior research on interventions primarily focused on individual (types of) interventions. The broad Human-In-The-Loop framework (Cranor 2008) shows many factors that influence the effectiveness of security interventions, but remains on a descriptive level of theory: The framework only describes the factors, but does not guide the developers on how to arrive at an intervention.

To better support developers in future, we propose an intervention framework that relates the intervention strategy and content to the context and human factors of a corresponding situation as well as knowledge bases of human behavior and threats (cf. Figure 1). We further present a methodology to derive the relevant factors and knowledge bases, which employs user studies, literature reviews, and expert consultations. While applicable in various domains, we focus on Web browsing as one important application area when giving examples in this paper.

2. PRIOR RESEARCH ON SECURITY INTERVENTIONS

Among the areas that security-intervention researchers have focused upon is that of *intervention strategies*, that is, when and in which form to intervene. For example, Whalen and Inkpen (2005) showed how symbols as a passive form of interventions are seen, but not interacted with by the users. Wu et al. (2006) argued that the right timing is important for interventions. Generally, active warnings have been shown to be more effective than passive indicators (Schechter et al. 2007). However, overly frequent warnings (e.g. from false positives) lead to habituation effects (Amer and Maris 2006).

Further research occurred on the *content of interventions*. Bravo-Lillo et al. (2011) showed empirically that warnings are not understood – for example, due to technical terminology. Biddle et al. (2009) found that their reformulated warnings made users more responsive to different levels of connection security. Wogalter (2006) argues that warnings need to inform about or remind of the *threats and consequences*. Downs et al. (2006) showed that phishing warnings are more often ignored if the threats and consequences are unknown. Furthermore, Kauer et al. (2012) found that individuals are more likely to heed warnings if they perceive personal consequences.

Wogalter (2006) also argues that warnings need to fit the audience and that *personal characteristics* should be taken into account when designing security indicators. Lin et al. (2011), for example, found that domain highlighting helps a subset of their study participants, depending on the participants' expertise. Bravo-Lillo et al. (2011) apply the Human-in-the-Loop framework (Cranor 2008) to warnings to describe the various factors that influence the

Indicator	Scope	Measurement
Trustworthiness of operator	Web site	Recommendations
Connection encryption	Connection	Browser
User expertise	User	Questionnaire

Table 1: *Examples of context indicators in Web browsing*

behavior of the user – for example, they show that behavior depends on expertise and prior experience.

To warn in an adequate form and achieve the necessary impact, De Keukelaere et al. (2009) proposed to adapt the intervention to the *context*; they found improvements from considering the security risk and prior actions of the user.

3. SECURITY INTERVENTION FRAMEWORK

The goal of the framework is to support the development of suitable security interventions. The literature in the previous section points to the primary concepts of the framework, depicted with their most important interrelations in Figure 1. The outcome for the developer is whether, when, and in which form the intervention appears (*intervention strategy*, e.g. active as a warning or passive as a symbol), and what *content* it conveys (e.g. technical threats or personal consequences).

Both the appearance and the content of the intervention is in our framework primarily influenced by the *context* and *personal characteristics* of the user. Context indicators measure the security and further aspects of the context, and vary concerning scope and type of measurement (see Table 1 for examples from Web browsing).

The information about the context needs to be interpreted and modulated according to two knowledge bases. The first concerns *human behavior*, particularly the *trust decisions* – e.g. under which circumstances users will trust a Web site enough to enter a password (green location bar, professional design, user expertise). Combining the information from the context (e.g. whether the location bar is green) and the knowledge on trust decisions will allow us to estimate the behavior of the current user and to what degree the user needs to be influenced through an intervention. Herein, we take the existing trust signals (e.g. green location bar) as the base line.

As the second knowledge base, the structure of relevant *threats and consequences* enables a more effective formulation of the content of the intervention. By interrelating context indicators, threats, and consequences, the specific consequences relevant to the situation can be identified. In combination with the knowledge on trust decisions, those threats and consequences can be selected that are considered most effective for the user (e.g. those with highest personal value or those unaware of), depending on her experience and expertise.

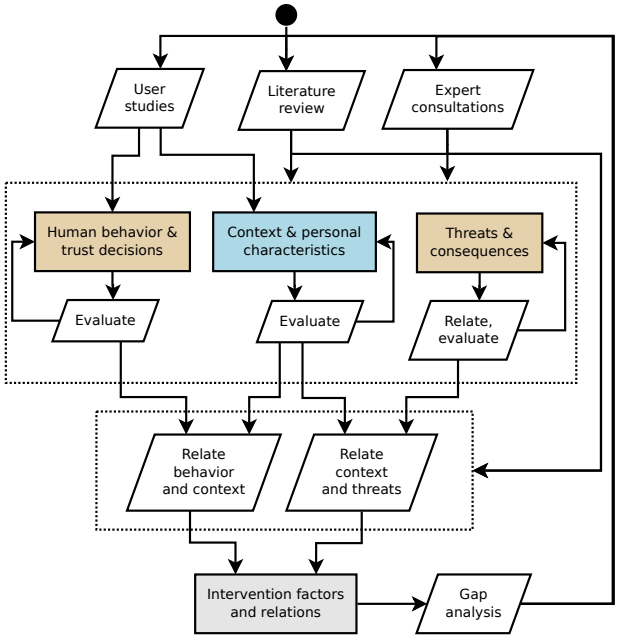

Figure 2: *Intervention-factor elicitation methodology*

4. ELICITATION OF INTERVENTION FACTORS

To operationalize the framework, developers need to elicit the intervention factors and build the knowledge bases for their application area (e.g. Web browsing). We propose the methodology depicted in Figure 2, which employs expert consultations, literature reviews, and user studies to elicit the factors (context indicators, factors in human behavior, and threats and consequences). Context indicators are then evaluated for their availability (e.g. how can we elicit the user's Web browsing intention). Human-behavior factors and threats and consequences are evaluated for their influence on the intervention. These sources are also used to interrelate context indicators and factors to derive graphs for the decisions on the intervention strategy and the dynamic construction of intervention content: How threats may lead to specific consequences, and how context indicators signal specific behavior and threats. Lastly, a gap analysis is performed based on the graphs (e.g. for missing context indicators to identify specific threats) and may trigger an additional iteration.

Not all of these activities will be necessary in their entirety for each newly developed intervention. We foresee general and domain-specific knowledge bases, which, for example, are provided by researchers and which developers then tailor to the specific application.

5. DISCUSSION AND FUTURE WORK

We collected first experiences on applying the methodology in two application areas: Contextualized warnings for Web browsing, which, amongst other aspects, consider the intentions of the user; and for email communication, e.g. addressing malicious email content (phishing) and attachments. In both cases, we conducted literature reviews for human behavior (to identify how indicators influence users), context indicators (what indicators exist and how they can be measured), and threats (which threats and consequences exist in the application). In addition, we applied expert consultations for threat analysis (how are the various threats and consequences interrelated and how do they relate to context indicators) and user studies (how can we elicit the intentions of users in Web browsing as a context indicator). By applying the methodology, we could already derive promising dynamic warnings for concrete situations.

The primary goal for future work is to evaluate the practical applicability of the methodology on two levels: regarding the resulting intervention (efficiency and effectiveness of the intervention for the users) and regarding the development process for the developer. Since the evaluation of generative theories is generally challenging (experimental settings are difficult), we will analytically evaluate the developer effort as the first step and conduct user studies on the resulting interventions. The evaluation will also include the level of complexity of the factors that is necessary to arrive at superior interventions. Lastly, we will study in which ways we can generalize knowledge bases and algorithms that build upon the framework to derive interventions.

REFERENCES

T.S. Amer and J.B. Maris. Signal Words and Signal Icons in Application Control and Information Technology Exception Messages – Hazard Matching and Habituation Effects. Technical Report 06-05, Nothern Arizona University, 2006.

R. Biddle, P. C. van Oorschot, A.S. Patrick, J. Sobey, and T. Whalen. Browser interfaces and extended validation SSL certificates: an empirical study. In *Proceedings of the 2009 ACM workshop on Cloud computing security*, CCSW '09, pages 19–30, New York, NY, USA, 2009. ACM.

R. Böhme and J. Grossklags. The security cost of cheap user interaction. NSPW '11, pages 67–82, New York, NY, USA, 2011. ACM.

C. Bravo-Lillo, L.F. Cranor, J.S. Downs, and S. Komanduri. Bridging the Gap in Computer Security Warnings: A Mental Model Approach. *IEEE Security & Privacy*, 9(2):18 – 26, Mar–Apr 2011.

L.F. Cranor. A Framework for Reasoning About the Human in the Loop. In *UPSEC 08*, Pittsburgh, Pennsylvania, 2008.

F. De Keukelaere, S. Yoshihama, S. Trent, Y. Zhang, L. Luo, and M. Zurko. Adaptive Security Dialogs for Improved Security Behavior of Users. In Tom Gross, et al., editors, *Human-Computer Interaction – INTERACT 2009*, volume 5726 of *Lecture Notes in Computer Science*, pages 510–523. Springer Berlin / Heidelberg, 2009.

J.S. Downs, M.B. Holbrook, and L.F. Cranor. Decision strategies and susceptibility to phishing. In *SOUPS '06: Proceedings of the second symposium on Usable privacy and security*, pages 79–90, New York, NY, USA, 2006. ACM.

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 *GI SICHERHEIT 2012 Sicherheit – Schutz und Zuverlässigkeit*, 2012.

E. Lin, S. Greenberg, E. Trotter, D. Ma, and J. Aycock. Does domain highlighting help people identify phishing sites? CHI '11, pages 2075–2084, New York, NY, USA, 2011. ACM.

S.E. Schechter, R. Dhamija, A. Ozment, and I. Fischer. The Emperor's New Security Indicators. In *S&P '07: Proceedings of the 2007 IEEE Symposium on Security and Privacy*, pages 51 – 65, May 2007.

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

T. Whalen and K.M. Inkpen. Gathering evidence: use of visual security cues in web browsers. GI '05, pages 137–144, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2005. Canadian Human-Computer Communications Society.

M.S. Wogalter. *Handbook of warnings*. Routledge, 2006.

M. Wu, R.C. Miller, and S.L. Garfinkel. Do security toolbars actually prevent phishing attacks? In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 601–610, New York, NY, USA, 2006. ACM.