Engaging stakeholders during late stage security design with assumption personas

Shamal Faily
Department of Computing and Informatics, Bournemouth University, Poole, UK

Received 13 October 2014
Revised 11 January 2015
Accepted 12 January 2015

Abstract

Purpose – This paper aims to present an approach where assumption personas are used to engage stakeholders in the elicitation and specification of security requirements at a late stage of a system's design.

Design/methodology/approach – The author has devised an approach for developing assumption personas for use in participatory design sessions during the later stages of a system's design. The author validates this approach using a case study in the e-Science domain.

Findings – Engagement follows by focusing on the indirect, rather than direct, implications of security. More design approaches are needed for treating security at a comparatively late stage. Security design techniques should scale to working with sub-optimal input data.

Originality/value – This paper contributes an approach where assumption personas engage project team members when eliciting and specifying security requirements at the late stages of a project.

Keywords Information security, Business analysis, Software engineering

Paper type Research paper

1. Introduction

When building software, security is considered as an after-thought, and security requirements are not properly considered until a comparatively late stage. When eliciting security requirements, stakeholders need to be engaged to provide insights into potential vulnerabilities and threats, but this can be difficult. The right stakeholders may be heavily in demand, and motivated by innovation rather than security. Stakeholders dedicate significant time and resources to understanding the complexity of a problem domain, leaving themselves little time for engaging with standard security design techniques. Such stakeholders may also find security a distant topic, with media reports on security threats and privacy invasions as somehow irrelevant to a system they are trying to build.

One way of engaging the security unengaged is to rely not only on evocation, but also people's natural bias towards personified, rather than anonymous, risk (Schneier 2012).

The research described in this paper was funded by EPSRC CASE Studentship R07437/CN001. The author is very grateful to QinetiQ Ltd for their sponsorship of this work.

Information & Computer Security
Vol. 23 No. 4, 2015
pp. 435-446
Emerald Group Publishing Limited
2056-4961
DOI 10.1108/ICS-10-2014-0066

Software developers may gloss over stories about the loss or public disclosure of patient medical data, but highlighting *their* contribution towards such losses may draw their attention. User experience (UX) artefacts can evoke by contextualising or personifying these losses, but building them requires real-world empirical data; this can only be collected when stakeholders are engaged, thereby leading to a "chicken-and-egg" situation.

Recent work (Dray 2014) has highlighted the power of assumptions towards engaging developers. Using UX research to challenge assumptions helps developers recognise why such issues need to be addressed, and focus their curiosity towards addressing them. To explore the power of challenging such assumptions, this paper presents an approach for eliciting and specifying security requirements using assumption-based personas, scenarios and risks to engage system developers to think more about security for a medical research portal, particularly how the portal might be misused. Section 2 briefly describes the related work upon which this approach is based, before presenting the approach in Section 3. Section 4 describes some results of applying the approach in a case study, before concluding with some lessons learned for security design in Section 5.

2. Related work

2.1 Personifying security expectations

The *personas* technique is a popular UX approach for personifying users to understand their goals and needs (Cooper *et al.*, 2007). Personas were designed to provide a specification of archetypical users, enabling software developers to design software to satisfy their expectations, rather than relying on their own assumptions about users, which may be unwarranted. In recent years, personas have also been used to support secure system design interventions. For example, Faily and Fléchais (2010) found that personas not only provided empathy about the security challenges of hard-to-reach user groups, but were useful for eliciting unforeseen user characteristics if stakeholders felt a persona did not match reality.

Personas are data-driven, and collecting the empirical data necessary to build them is difficult if stakeholders are not engaged enough to provide or facilitate access to such data. Given these difficulties, Pruitt and Adlin (2006) propose the use of *assumption personas*; these are sketches that articulate assumptions about a user population. Once created, assumption personas allow stakeholders to see the value of personas, and how assumptions may colour their characteristics.

2.2 Contextualising personas in secure system design

Personas build empathy, but their goals and expectations need to be put in context. For this reason, personas are often paired with scenarios; these centre around activities performed by users, rather than around the users themselves. For example, Rosson and Carroll (2002) used scenarios to describe how hypothetical stakeholders tackle current practice; these scenarios may be based on empirical data or assumptions. More recently, Parkin *et al.* (2010) successfully engaged senior managers using low-fidelity prototypes of security management tools, and a collection of scenarios illustrating their use. Together, personas and scenarios illustrate security problems, but understanding these problems is not enough to specify solutions that address them: we need to carry out a more formal security and requirements analysis. Personas and scenarios supplement these analyses by illustrating how risks are realised, and how specification decisions are operationalised. In doing so, the human implications of security design decisions in different contexts of use can be better perceived.

2.3 Integrating requirements and information security

To explore how these different approaches to design in security, usability and requirements engineering might be used together to design secure and usable systems, the integrating requirements and information security (IRIS) process framework was devised (Faily 2011). IRIS demonstrates how the elements constituting personas, scenarios, requirements and risks might be aligned, and the application of security, usability and requirements techniques can complement each other. The framework provides guidance for instantiating design processes; these constitute ordered collections of design techniques, informed by the context within which they are applied. The techniques in a process should elicit all the concepts stipulated by IRIS' meta-model (Faily and Fléchais, 2010a).

IRIS is tool-supported by the open-source computer-aided integration of requirements and information security (CAIRIS) requirements management tool (Faily, 2013). It allows the capture of security, usability and requirements data as design techniques are applied; guides the creation of personas; and automatically evaluates risks for different contexts of use (Faily and Lyle, 2013; Faily and Fléchais, 2010b).

The application of personas in IRIS helps contextualise different aspects of a system's specified design. Moreover, because the data elicitation and analysis activities associated with persona creation can be re-used in complementary design techniques, the framework is useful when design activities need to be scheduled at short notice. This was illustrated by Faily and Fléchais (2011), where IRIS was used to analyse a critical infrastructure organisation's security policy in response to reports of the Stuxnet worm.

3. Approach

Using personas and scenarios, an approach for eliciting and specifying security requirements that engages stakeholders during the late stages of a system's design has been developed. This approach, which is an instantiated IRIS process, entails creating personas grounded in assumptions in design documentation about a system's users. These personas are used to contextualise scenarios, and build asset and goal models that, together, consider different aspects of how a system is used, or inadvertently misused. Requirements are elicited during these design sessions, and specified and managed within CAIRIS.

This approach, which is summarised in Figure 1, not only captures information about how usability and security concerns impact requirements, it also accommodates a lack of end-user access, limited access to project stakeholders and the need to make assumptions about users as transparent as possible during design.

3.1 Assumption persona development

The first stage of the approach involves specifying the expectations held about a system's prospective user-community. Implicit assumptions in the available documentation are identified, and used to form the basis of assumption personas. Not only do these assumption personas clarify expectations about end-users, subsequent discussion around these confirm a useful scope of analysis for the subsequent stage.

For each role relevant to the scope of analysis, the available documentation is reviewed to elicit factoids for each role. These factoids are structured using argumentation models (Toulmin, 2003) to provide a basis for validating the assumptions underpinning personas. Each persona characteristic is aligned with a claim made. Propositions about assumptions made about characteristics may act as *grounds* of evidence, or a *warrant* describing how the grounds contribute to the claim. The origin of a warrant's assumption is the *backing* knowledge for believing the claim. Finally, a *modal qualifier* indicates the degree of certainty about the claim. Based on these characteristics, narratives are written. Once the assumption personas are developed, these are presented to the project team for review. Any issues raised by the team are used to revise the assumption personas or correct any misinterpretations held about the system. The process for building these personas is described in more detail by Faily and Fléchais (2010b).

3.2 Design sessions

This stage entails holding small focus groups with project team members. Each session focuses on the use of scenarios, requirements or risk analysis.

A scenario session involves modelling scenarios carried out by the assumption personas in their respective contexts. Like the personas, these scenarios are grounded in assumptions identified from project documentation, or from analysis undertaken during other design sessions. Some of these scenarios focus on *misusability*, by illustrating how unintentional misuse of the system might lead to security problems.

A requirements session involves using the KAOS goal-oriented requirements engineering approach (van Lamsweerde, 2009) to elicit and specify requirements needing to be satisfied for the scenarios to be realised. Requirements are modelled as goal trees and, in addition to being refined to sub-goals, goals may conflict with obstacles: conditions representing undesired behaviour and preventing an associated goal from being achieved (van Lamsweerde and Letier, 2000). Such obstacles may arise from intentional, as well as accidental, misuse, thereby making it possible for them to model threats (van Lamsweerde, 2004).

Risk analysis sessions involve using AEGIS (appropriate and effective guidance for information security): a participative design process (Fléchais *et al.*, 2007). This entails the team members jointly modelling the system's assets in different contexts; these assets are modelled using UML class diagrams, where classes represent different assets. The assets are evaluated according to values held by the participants about them.

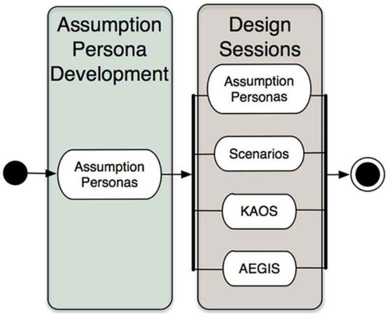

Figure 1.
UML activity diagram describing an approach for late-stage requirements elicitation

Vulnerabilities, threats and risks affecting these assets are elicited, before possible security controls mitigating these risks are selected. Although one of many risk analysis processes, AEGIS's diagrammatic notation is useful for engaging stakeholders about security, providing useful discussion about asset values and eventually yielding relevant security requirements (Fléchais, 2005).

In all sessions, assumption personas are used as an authority for user expectations; these are modified if aspects of the analysis challenge their characteristics.

During each session, elicited requirements and security analysis elements are specified within CAIRIS, and the resulting models are discussed with the session participants. After the final session, each requirement is examined and assigned a responsible role. Following this, a specification document is generated and sent to project team members for review.

4. Case study

This approach was evaluated by using it to elicit security requirements for a portal that facilitated the sharing of medical study data. The study data consisted of long-running, longitudinal studies of people sharing some specific characteristic. By providing an accessible interface to such studies, the portal ensured that research data and metadata could be re-used by researchers, thereby reducing the need for running unnecessary and expensive long-term studies.

4.1 Study factors

The study began with a kick-off meeting with the stakeholders to learn more about the context within which the project operated. The author also had the opportunity to observe a half-day project progress meeting, attended by team members. From both these observations and background research, the following factors were identified as likely to have an impact on the study.

4.1.1 E-Science and security. An implicit objective of the portal was to leverage technology and data towards better medical research. As such, the project also furthered research in the area of *e-Science*, which is concerned with the global collaboration in key areas of science and the next generation of infrastructure that will enable it (Taylor, 2001).

Although some of the design rationale underpinning the portal was motivated by previous research (Crichton *et al.*, 2009), there was sufficient novelty in both the domain and the implementation technology to make invention, rather than quality, the team's primary concern. Although one aim of e-Science research is to cast light on some of these uncertainties, securing innovation tends not to be treated as a priority area. Work is beginning to address the challenge of securing e-Science activities (Martin *et al.*, 2010); however, e-Science may be one developmental climate where invention will always be prioritised over quality.

Prioritising core functionality does not mean that security is ignored in e-Science. Rather, there is, as Martin *et al.* (2010) suggest, a tendency to treat it in an *ad hoc* manner. Given the different perceptions stakeholders might hold about assets in an e-Science project, security design decisions might be over- or under-commensurate with assets needing to be safeguarded. For example, at one level, highly aggregated data sources may not appear to be a valuable target for an attacker, although, with the right search criteria, it is possible to de-anonymise data sources based on the criteria used to search data.

Without careful thought about e-Science assets, and the threats and vulnerabilities associated with them, security decisions may also have an unpredictable impact on the project's user community. This means that the user communities associated with e-Science projects have characteristics which need to be considered as part of any intervention.

4.1.2 The user community. Two user roles dominated the design of the portal. The first of these were academic researchers; these would use the portal to find data sets of interest. The project sponsors were keen to maximise take-up by the researcher community, and initiatives encouraging this would be looked upon kindly. The second class of users were data managers; these were responsible for curating data sets, which would be available via the portal. The perception held by the project team was that data managers were the portal's key user community.

The portal design was dominated by two contexts of use. The *Research* context was concerned with researchers interacting with the portal as part of their day-to-day research. The *Study* context was concerned with data managers' interaction with the portal to curate their data sets, and managing requests for accessing them.

4.1.3 Stakeholder access. Although empirical data from representative stakeholders would have made an invaluable contribution to this study, there was no scope for collecting data from research end-users. Similarly, time constraints meant that data managers from the study exemplars would also not be available. Fortunately, the development team agreed to act as user proxies because of the time they had spent working with data managers, and their domain knowledge based on previous, related research. However, given that the development team had little direct experience of the researcher community, it was important that any assumptions that were made about both data managers and researchers were as transparent as possible.

Unfortunately, limited access to stakeholders also extended to the development team. The project team was small with only four developers, and faced several tight deadlines; this severely limited opportunities for working with the project team. Consequently, it was essential to be parsimonious with regards to team member access, while at the same time ensuring that the study had an impact on the development of the portal.

4.2 Assumption persona development

When the study began, only two documents were available for eliciting assumptions: a requirements specification for the portal, and a technical annexe to the portal's contract signed by all project partners.

After a review of the documentation, three roles were evident: researchers, data managers and gateway administrators. Based on these roles, the documentation was analysed to elicit assumptions. Assumptions were elicited about behaviour, which could be reasonably assumed if the documentation accurately represented the concerns of the particular stakeholder role. For example, a requirement indicating that first-line support to the portal would be provided between 9 a.m. and 5.30 p.m. might reasonably suggest that the authors believe researchers work only during commercial office hours. Three skeleton assumption personas were created for each of these roles: Alex (an academic researcher), Brian (a data manager) and Colin (an administrator for the data gateway). For each persona's characteristic, an argumentation model was constructed with a commensurate narrative. For example, based on persona characteristics summarised by *In no hurry* and *Looking to apply data-set once discovered*, the following narrative describing Alex's motivation was written:

> Alex is looking to use a dataset as soon as he discovers it is suitable. He isn't in a particular hurry, so is prepared to wait for his registration to the Data Gateway and the respective data set to be approved.

The results of this initial analysis to date were presented to the project team. This presentation described the scope assumed for the analysis, provided an overview of the work carried out, and Alex, Brian and Colin were presented. Each persona selected particular persona characteristics. The third characteristic was chosen as the most divisive, to stimulate lively discussion about the persona. For example, the bullet points below summarised some characteristics of Alex:

- frequently re-uses data between studies;
- Googles for ideas and data; and
- interested in study policies, data curatorship and happy to use a prototypical data gateway.

In this example, there was disagreement among the developers whether Alex would use Google, and the merits of using a general search engine to find study data. This characteristic was identified based on portal requirements indicating that the portal would need to be search engine-optimised. Based on the discussion, the developers agreed that it would be more likely that Alex would use PubMed (National Center for Biotechnology Information, 2014) as a search engine instead. As a result, the argumentation model underpinning the relevant characteristic was updated to reflect this; this updated model is illustrated in Figure 2.

At the end of this session, it was agreed that Colin's activities were not relevant to the scope of analysis, and this persona was dropped from the remainder of the study.

Despite the nature of the documentation used, it was possible to elicit a surprising amount of data about both the possible activities and attitudes of personas. Moreover, many persona characteristics were elicited during the design sessions. As such, the personas evolved throughout the design sessions, concurring with best practice in the use of personas, which suggests that personas should be fostered throughout a project's life cycle (Pruitt and Adlin, 2006).

Although identifying the basis for characteristics was straightforward, justifying them was not. Prior to their initial validation, many of the characteristics were based exclusively on individual pieces of empirical data. As such, value judgements about the source data and context were directly reflected in these characteristics.

Although the initial workshop surfaced a number of these issues, it was usually not until the personas were directly written into scenarios in design sessions that certain invalid characteristics were identified. Applying the personas within a specific context did, however, help identify missing data about behaviour not identified during their creation and initial presentation to the team.

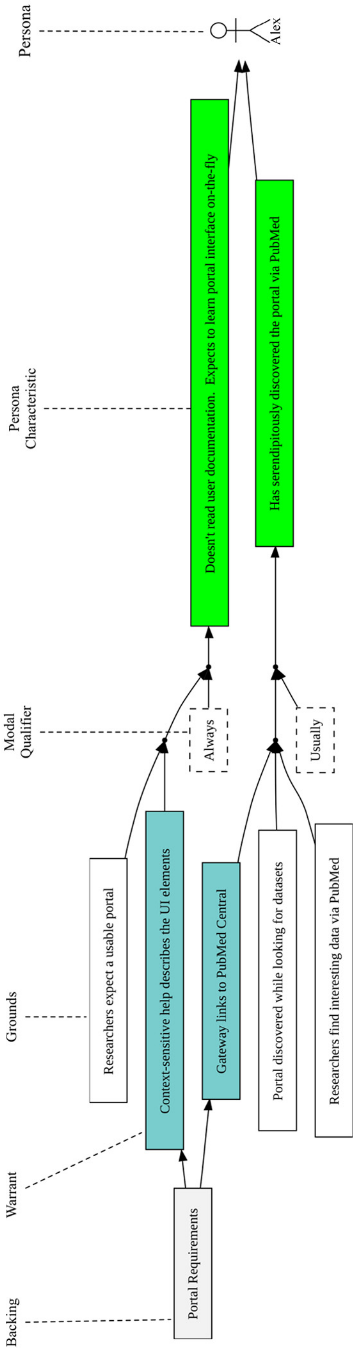
Persona
Alex
Persona Characteristic
Doesn't read user documentation. Expects to learn portal interface on-the-fly
Has serendipitously discovered the portal via PubMed
Modal Qualifier
Always
Usually
Grounds
Researchers expect a usable portal
Context-sensitive help describes the UI elements
Gateway links to PubMed Central
Portal discovered while looking for datasets
Researchers find interesting data via PubMed
Warrant
Backing
Portal Requirements

Figure 2.
Argumentation model underpinning two of Alex's characteristics

4.3 Design sessions

Four design sessions were held over the course of one month. The first was a scenario session, followed, a day later, by a requirements session. A risk analysis session took place the following week, followed by a final risk analysis session the week after.

Due to project deadlines, rather than having access to multiple developers per session, only a single developer was available. The same developer was consistently used for each session, and was available for email clarification when queries arose outside at other times. In addition to the allocated portal developer, a non-project domain expert participated in the second session. Although this participant was only partially aware of the on-going project, she was aware of the problem domain in general.

The approach taken during each session was more flexible than originally envisaged. In practice, multiple techniques were used when the situation deemed it useful. For example, KAOS was used in each of the first sessions when it was felt most appropriate. Similarly, elements of AEGIS were also used in the first two sessions to elicit assets, their relationships and concerns arising from goals and scenarios. Switching from the use of one technique to another did not seem to hinder the thought processes of participants during the sessions.

The resulting AEGIS asset models were automatically re-generated based on information entered into CAIRIS. These models helped participants contextualise some of the ways that personas interact with assets. An example of such an asset model is shown in Figure 3.

During the sessions, the personas were progressively refined and embellished with further characteristics from the documentation as new insights were gleaned.

The amount of data elicited from the risk analysis sessions was comparatively small. This was mainly due to the resolution of many problems during the requirements and scenarios sessions. Another reason for the small number of explicit risks was a tendency by the project team to dismiss security issues deemed out of scope. On more than one occasion, assets identified as in-scope, such as portal documentation about the use of some functionality, were de-scoped. This issue of passing responsibility for out-of-scope issues was also apparent from the threats and vulnerabilities highlighted in both contexts of use.

The issue of risk deferral was also contextual. Most of the risk analysis elements were elicited from the research context; these were associated with assets deemed out of the project scope. The few risk analysis elements not concerning the study environment were also marginalised. For example, of four risks elicited, only one – a Man-in-the-Middle attack – concerned the study environment. Upon discussing resolutions to this, it was agreed that the portal relied on a secure channel between some of its components. Consequently, responsibility for mitigating more general Man-in-the-Middle attacks was delegated to the administration team responsible for one of these components.

Although the project team were reluctant to take a defence-in-depth approach to tackling security problems, security concerns were eventually identified. This was possible by drawing attention to goal obstruction within the study environment; unlike the research environment, this environment concerned concepts that were within the project scope. This allowed threats and vulnerabilities to be mitigated at the design level when considered in context with other portal requirements. This was especially useful

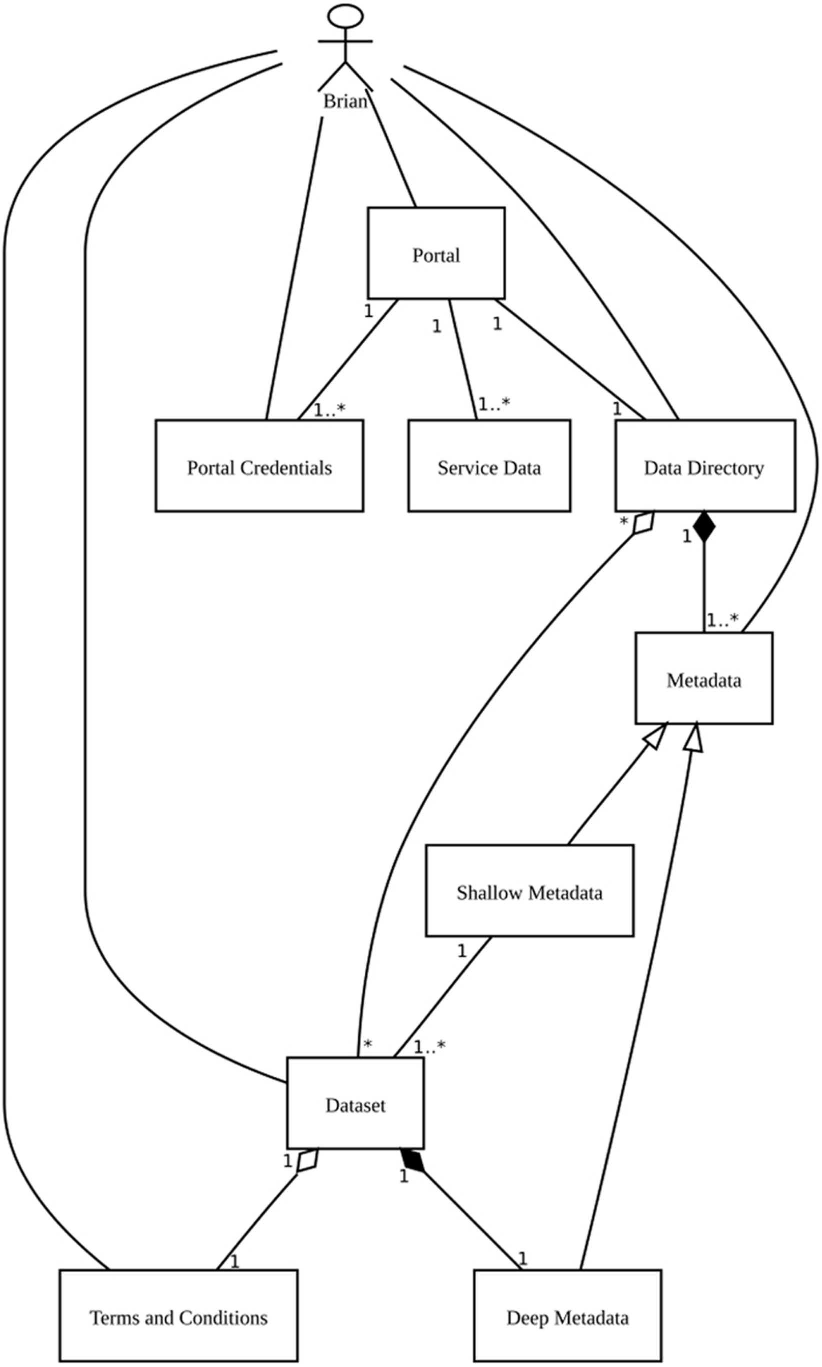

Figure 3.
CAIRIS-generated AEGIS asset model of the study context

because, besides the generic Internet-facing threats and vulnerabilities, it was not entirely clear what the threat model facing the portal might be.

5. Conclusion

This paper presented an approach for engaging stakeholders in the elicitation and specification of security requirements at a late stage of a system's design. Based on the lessons learned applying this approach, there are three security design lessons that can be taken away.

First, engagement can follow by focusing on the indirect, rather than direct, implications of security. One of the difficulties in completing this study arose from the lack of engagement with the project team. Although the project team appeared to be genuinely interested in the approach and the analysis being carried out, their time was too limited to properly integrate this analysis into the day-to-day running of the project. The study showed that focusing on the impact of non-security design decisions is a more effective technique for engaging developers in security issues rather than relying on fear of more generic threats, particular when these threats may or may not be relevant (or perceived relevant) to the scope of analysis.

Second, as much as security should be considered at the outset of a project, design approaches for treating it at a comparatively late stage are not infeasible. This is an important finding, as the individual techniques used were designed on the basis that they would be used early in the design process. In contrast, not only were they applied comparatively late, they were used in parallel with other design activities.

Finally, our findings lead us to conclude that, when a security design process is devised, its techniques should scale to working with less-than-optimal input data. Moreover, the process should attempt to carry out as much analysis as reasonably possible without disrupting other project activities.

References

Cooper, A.R. and Cronin, D. (2007), *About Face 3: The Essentials of Interaction Design*, John Wiley & Sons, Hoboken, NJ.

Crichton, C., Davies, J. Gibbons, J. Harris, S. Tsui, A. and Brenton, J. (2009), "Metadata-driven software for clinical trials", *Proceedings of the 2009 ICSE Workshop on Software Engineering in Health Care*, *IEEE Computer Society*, pp. 1-11.

Dray, S.M. (2014), "Questioning assumptions: UX research that really matters", *Interactions*, Vol. 21 No. 2, pp. 82-85.

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

Faily, S. (2013), "CAIRIS web site", available at: http://github.com/failys/CAIRIS (accessed 13 October 2014).

Faily, S. and Fléchais, I. (2010), "Barry is not the weakest link: eliciting secure system requirements with Personas", *Proceedings of the 24th BCS Interaction Specialist Group Conference, BCS '10*, *British Computer Society*, pp. 124-132.

Faily, S. and Fléchais, I. (2010a), "A meta-model for usable secure requirements engineering", *Proceedings of the 6th International Workshop on Software Engineering for Secure Systems*, *IEEE Computer Society*, pp. 126-135.

Faily, S. and Fléchais, I. (2010b), "The secret lives of assumptions: developing and refining assumption personas for secure system design", *Proceedings of the 3rd Conference on Human-Centered Software Engineering*, *Springer*, pp. 111-118.

Faily, S. and Fléchais, I. (2011), "User-centered information security policy development in a post-Stuxnet world", *Proceedings of the 6th International Conference on Availability, Reliability and Security*, pp. 716-721.

Faily, S. and Lyle, J. (2013), "Guidelines for integrating personas into software engineering tools", *Proceedings of the 5th ACM SIGCHI Symposium on Engineering Interactive Computing Systems, EICS '13, ACM*, pp. 69-74.

Fléchais, I. (2005), "Designing secure and usable systems", PhD thesis, University College, London.

Fléchais, I., Mascolo, C. and Sasse, M.A. (2007), "Integrating security and usability into the requirements and design process", *International Journal of Electronic Security and Digital Forensics*, Vol. 1 No. 1, pp. 12-26.

Martin, A., Davies, J. and Harris, S. (2010), "Towards a framework for security in e-Science", *IEEE E-Science 2010 Conference, Oxford University*, *Oxford*.

National Center for Biotechnology Information. (2014), "PubMed.gov", available at: www.ncbi.nlm.nih.gov/pubmed (accessed 13 October 2014).

Parkin, S., van Moorsel, A. Inglesant, P. and Angela, S.M. (2010), "A stealth approach to usable security: helping IT security managers to identify workable security solutions", *Proceedings of the 2010 Workshop on New Security Paradigms, NSPW '10*, ACM, pp. 33-50.

Pruitt, J. and Adlin, T. (2006), *The Persona Lifecycle: Keeping People in Mind Throughout Product Design*, Elsevier, New York, NY.

Rosson, M.B. and Carroll, J.M. (2002), *Usability Engineering: Scenario-Based Development of Human-Computer Interaction*, Academic Press, Salt Lake City UT.

Schneier, B. (2012), *Liars & Outliers: Enabling the Trust That Society Needs to Thrive*, John Wiley & Sons, Hoboken, NJ.

Taylor, J. (2001), "Presentation at e-science meeting by the director of the research councils, office of science and technology, UK", available at: www.nesc.ac.uk/nesc/define.html (accessed 13 October 2014).

Toulmin, S. (2003), *The Uses of Argument*, Cambridge University Press, Cambridge.

van Lamsweerde, A. (2004), "Elaborating security requirements by construction of intentional anti-models", *Proceedings of the 26th International Conference on Software Engineering, IEEE Computer Society*, pp. 148-157.

van Lamsweerde, A. (2009), *Requirements Engineering: From System Goals to UML Models to Software Specifications*, John Wiley & Sons, Hoboken, NJ.

van Lamsweerde, A. and Letier, E. (2000), "Handling obstacles in goal-oriented requirements engineering", *IEEE Transactions on Software Engineering*, Vol. 26 No. 10, pp. 978-1005.

About the author

Shamal Faily is a Lecturer in Systems Security Engineering at the Department of Computing & Informatics at Bournemouth University. His research explores how interaction design techniques and software tools can be used to build and maintain systems that are not only secure, but also usable within different contexts of use. Shamal Faily can be contacted at: sfaily@bournemouth.ac.uk