Eliciting Policy Requirements for Critical National Infrastructure Using the IRIS Framework

Shamal Faily, University of Oxford, UK
Ivan Fléchais, University of Oxford, UK

ABSTRACT

Despite existing work on dealing with security and usability concerns during the early stages of design, there has been little work on synthesising the contributions of these fields into processes for specifying and designing systems. Without a better understanding of how to deal with both concerns at an early stage, the design process risks disenfranchising stakeholders, and resulting systems may not be situated in their contexts of use. This paper presents the IRIS process framework, which guides technique selection when specifying usable and secure systems. The authors illustrate the framework by describing a case study where the process framework was used to derive missing requirements for an information security policy for a UK water company following reports of the Stuxnet worm. The authors conclude with three lessons informing future efforts to integrate Security, Usability, and Requirements Engineering techniques for secure system design.

Keywords: Computer-Aided Integration of Requirements and Information Security (CAIRIS), Integrating Requirements and Information Security (IRIS), Knowledge Acquisition in automAted Specification (KAOS), Misuse Cases, Personas

1. INTRODUCTION

There is no longer any obvious reason why designing secure and usable systems should be so difficult, especially when guidance on applying Security and Usability Engineering best practice is no longer restricted to the scholarly literature. Several years ago, Nielsen claimed that cost was the principal reason why Usability Engineering techniques are not used in practice (Nielsen, 1994), but technology advances have reduced the financial costs of applying such techniques. Similarly, practical techniques for identifying and mitigating security problems during system design are now available to developers in an easy to digest format (e.g., Schneier, 2000; Swiderski & Snyder, 2004).

Problems arise when considering how to use these approaches as part of an integrated process. Accepted wisdom in software engineering states that requirements analysis and specification activities should precede other stages in a project’s lifecycle (Ghezzi et al., 2003). However, Information Security and HCI proponents argue that their techniques should instead come first. For example, ISO 13407
(ISO, 1999) states that activities focusing on
the collection of empirical data about users
and their activities should guide early design,
but security design methods such as Braber et
al. (2007) suggest that such stages should be
devoted to high-level analysis of the system
to be secured. Invariably, the decision of what
concern to put first is delegated to the meth-
odology followed by a designer. The designer
has many approaches to choose from, some of
which include treatment for security or usability
concerns. To date, however, no approach treats
both security and usability collectively, beyond
treating them both as generic qualities contend-
ing with functionality.

The IRIS (Integrating Requirements and
Information Security) framework was first
introduced by the authors in Faily and Fléchais
(2009) to explore the challenges of designing
systems with both information security and
HCI in mind. This framework encompassed
three elements: a meta-model for usable secure
requirements engineering (Faily & Fléchais,
2010), a user-centered design method (illus-
trated in Faily & Fléchais, 2010), and comple-
mentary tool-support (Faily & Fléchais, 2010).
However, although the second element was
described as a method, this is more aptly defined
as a methodology. While a method describes
a concrete procedure for getting something
done, a methodology is a higher level construct
motivating the need for choosing between dif-
ferent methods (Iivari et al., 1998). Because
the terms method and methodology are used
interchangeably, the principles of information
system methodologies have been encapsulated
in several process frameworks that have, in
recent years, emerged in Software, Security,
and Usability Engineering. A framework can be
defined as a set of milestones indicating when
artifacts should be produced, as opposed to a
process describing the steps to be carried out
to produce the artifacts (Haley, 2007).

In this paper, we present the IRIS process
framework, which is used for selecting tech-
niques for specifying usable and secure systems.
Building on the meta-model described in Faily
and Fléchais (2010), we describe the different
perspectives of IRIS, and how IRIS concepts
and techniques are situated within these in
Section 3. We propose a number of exemplar
techniques for each perspective, and describe
modifications, which are necessary to situate
them within an IRIS process. In Section 4, we
describe how the IRIS process framework was
used to devise a user-centered approach for
eliciting information security policy require-
ments for a UK water company. The manage-
ment imperative for responding to the Stuxnet
worm (Control Engineering UK, 2010) meant
that policy decisions needed to be made where
there was both a lack of time for data collection
and restricted stakeholder availability. Finally,
in Section 5, we describe some of the lessons
learned carrying out this study, which, we
believe, inform future approaches for secure
system design.

2. RELATED WORK

Although frameworks exist for dealing with
security and usability as quality requirements
(e.g., Chung et al., 2004), we are unaware of
existing frameworks dealing explicitly with both
usability and security from a requirements per-
spective. There have, however, been processes
and frameworks purporting to deal with each.

2.1. Rescue

RESCUE (REquirements with SCenarios for a
User-centered Environment) is a user-centered
Requirements Engineering process (Maiden &
Jones, 2004). Although not explicitly defined as
a framework, the earlier phases of RESCUE af-
ford leeway in technique application. RESCUE
consists of the following four concurrent system
engineering streams: Human Activity Model-
ing, i* system modelling (Yu, 1995), Use Case
and Scenario Analysis (Cockburn, 2001), and
Requirements Management. Human Activity
Modelling involves analysing the way work
is carried out, and partitioning the analysis of
the problem domain into different aspects, such
as the work domain, control task, and social
organisation. When the system boundary has
Towards Deep Learning-Based Approach for Detecting Android Malware
[www.igi-global.com/article/towards-deep-learning-based-approach-for-detecting-android-malware/236204?camid=4v1a](www.igi-global.com/article/towards-deep-learning-based-approach-for-detecting-android-malware/236204?camid=4v1a)

Applying AHP for Collaborative Modeling Evaluation: Experiences from a Modeling Experiment
[www.igi-global.com/article/applying-ahp-collaborative-modeling-evaluation/75462?camid=4v1a](www.igi-global.com/article/applying-ahp-collaborative-modeling-evaluation/75462?camid=4v1a)

Toward a Statistical Characterization of Computer Daihinmin
[www.igi-global.com/article/toward-a-statistical-characterization-of-computer-daihinmin/217393?camid=4v1a](www.igi-global.com/article/toward-a-statistical-characterization-of-computer-daihinmin/217393?camid=4v1a)