Chapter 15
Towards Tool-Support for Usable Secure Requirements Engineering with CAIRIS

Shamal Faily
University of Oxford, UK

Ivan Fléchais
University of Oxford, UK

ABSTRACT

Understanding how to better elicit, specify, and manage requirements for secure and usable software systems is a key challenge in security software engineering, however, there lacks tool-support for specifying and managing the voluminous amounts of data the associated analysis yields. Without these tools, the subjectivity of analysis may increase as design activities progress. This paper describes CAIRIS (Computer Aided Integration of Requirements and Information Security), a step toward tool-support for usable secure requirements engineering. CAIRIS not only manages the elements associated with task, requirements, and risk analysis, it also supports subsequent analysis using novel approaches for analysing and visualising security and usability. The authors illustrate an application of CAIRIS by describing how it was used to support requirements analysis in a critical infrastructure case study.

INTRODUCTION

Frequent reports of human and technical security failures in systems highlight the need for designing usable security, but specifying usable and secure systems is easier said than done. Understanding why security controls are unusable means factoring in the characteristics of people using controls, the work they carry out while using controls, and the physical, social, and even cultural contexts within which the controls are used. While it is accepted wisdom that these concerns should be treated as

DOI: 10.4018/978-1-4666-1580-9.ch015

early as possible, eliciting and specifying requirements for secure and usable controls remains a hit-and-miss affair.

Requirements Engineering involves understanding the problem domain within which a system is situated, obtaining data from stakeholders in this domain, analysing this data to elicit a set of requirements, validating these requirements, and managing their evolution. When properly applied, techniques from HCI and Information Security complement these early stages of Requirements Engineering. Techniques used by usability professionals are grounded in observational, performance, and other qualitative and quantitative data. If used properly, this usability data can immerse analysts and stakeholders in the problem domain and help explore assumptions held about threats and vulnerabilities. Similarly, Goal-Oriented Requirements Engineering techniques are not only useful for eliciting requirements from goals, but also threats from anti-goals (Lamsweerde, 2004). Even the traditional workshop setting, where requirements are often elicited and validated, can support the design of usable security; participative approaches to risk analysis, e.g., (Fléchais et al., 2007; Braber et al., 2007) help stakeholders take a situated approach to security by alerting them to threats and vulnerabilities, identifying risks in their environment, and directing mitigating specification and design decisions.

The challenge of specifying usable and secure software systems comes not only from choosing the right combination of techniques, but also from analysing and managing the data arising from them. For non-trivial systems, risk and requirements analysis precipitate voluminous amounts of data. A requirement may be the leaf node of a large goal-tree, the root goal of which may be derived from mitigating a particular risk; this mitigation response may arise as a result of a chain of risk and requirements analysis. Furthermore, empirical usability data needs to contribute to any design decisions; if we mitigate one risk, the resulting usability impact of this design decision may introduce others. Risk and usability ratings for a system design are also coloured by analyst perceptions; this allows human error to creep into any valuation.

Without tool-support, the security-usability balance can become uneven and overly subjective as risk analysis becomes more advanced. We need tool-support to manage security, usability, and requirements data, automate its analysis, and convey the results to stakeholders. This paper discusses CAIRIS (Computer Aided Integration of Requirements and Risk Analysis): a tool for managing the elements arising from usability, requirements, and risk analysis. This tool supports the elicitation of requirements from goals and tasks, and risks from threats and vulnerabilities. By structuring elicited data according to a meta-model for usable secure requirements engineering (Faily & Fléchais, 2010), meaningful traceability links between different model types can be automatically maintained, allowing data to be quickly analysed and visualised in a participative workshop setting. In the next section, we describe the related work motivating CAIRIS. In the subsequent sections, we introducing the tool, and we describe how CAIRIS was used to elicit requirements in a Critical Infrastructure case study.

RELATED WORK

We are unaware of any single tool purporting to support the analysis of usability, requirements, and risk analysis. Some coverage is, however, provided by existing tools in each of these areas, and presented in the following sections.

Conceptual Tools for Usability

Designing usable system requires an early focus on users and their goals (Preece et al., 2007). Although many engineers consider usability as synonymous only with user interface design (Seffah & Metzker, 2004), it is also a quality concerning the people