Computer Standards & Interfaces 32 (2010) 153–165

Contents lists available at ScienceDirect

Computer Standards & Interfaces

journal homepage: www.elsevier.com/locate/csi

# A systematic review of security requirements engineering

Daniel Mellado [a], Carlos Blanco [b], Luis E. Sánchez [c], Eduardo Fernández-Medina [b,*]

[a] Spanish Tax Agency. Madrid, Spain
[b] Department of Information Technologies and Systems. University of Castilla-La Mancha. Alarcos Research Group. Paseo de la Universidad, 4. Ciudad Real, Spain
[c] SICAMAN Nuevas Tecnologías. Tomelloso, Ciudad Real, Spain

ARTICLE INFO

*Article history:*
Received 7 January 2009
Received in revised form 25 January 2010
Accepted 27 January 2010
Available online 2 February 2010

*Keywords:*
Security requirements
Security requirements engineering
Requirements engineering
Security engineering
Secure development
Security
Systematic review

ABSTRACT

One of the most important aspects in the achievement of secure software systems in the software development process is what is known as Security Requirements Engineering. However, very few reviews focus on this theme in a systematic, thorough and unbiased manner, that is, none of them perform a systematic review of security requirements engineering, and there is not, therefore, a sufficiently good context in which to operate. In this paper we carry out a systematic review of the existing literature concerning security requirements engineering in order to summarize the evidence regarding this issue and to provide a framework/background in which to appropriately position new research activities.



**Contents**

1. Introduction . . . 154
2. Question formalization . . . 155
   2.1. Question focus . . . 155
   2.2. Question quality and amplitude . . . 155
3. Review method . . . 155
   3.1. Sources selection . . . 155
   3.2. Studies selection . . . 156
   3.3. Selection execution . . . 156
4. Information extraction . . . 156
   4.1. Basin et al. "Model-driven security for process-oriented systems" [19] and "Model driven security: From UML models to access control infrastructures" [20] . . . 157
   4.2. Bresciani et al. "Tropos: Agent-Oriented Software Development Methodology" [21], Giorgini et al. "Requirements Engineering meets Trust Management: Model, Methodology, and Reasoning" [22] and Giorgini et al. "Modelling Security and Trust with Secure Tropos" [23], Ali et al. "Location-based Software Modeling and Analysis: Tropos-based Approach" [24] and "A Goal Modeling Framework for Self-Contextualizable Software" [25], Dalpiaz et al. "An Architecture for Requirements-Driven Selft-reconfiguration" [26], Massacci et al. "Using a security requirements engineering methodology in practice: The compliance with the Italian data protection legislation" [27] and Compagna et al. "How to integrate legal requirements engineering into a requirements engineering methodology for the development of security and privacy patterns" [28]. . . . 157
   4.3. Firesmith "Specifying Reusable Security Requirements" [6], "Engineering safety-related requirements for software-intensive systems" [29] and "Engineering Safety and Security Related Requirements for Software Intensive Systems" [30] . . . 157
   4.4. Hussein and Zulkernine "Intrusion detection aware component-based systems: A specification-based framework" [31] . . . 157
   4.5. Jennex "Modeling security requirements for information systems development" [32] . . . 157
   4.6. Lamsweerde, "Engineering requirements for system reliability and security" [65] . . . 157
   4.7. J. Lee et al. "A CC-based Security Engineering Process Evaluation Model" [33] . . . 157
   4.8. Lee et al. "Building problem domain ontology from security requirements in regulatory documents" [34] . . . 158

* Corresponding author.
*E-mail addresses:* damefe@esdebian.org (D. Mellado), Carlos.Blanco@uclm.es (C. Blanco), lesanchez@sicaman-nt.com (L.E. Sánchez), Eduardo.FdezMedina@uclm.es (E. Fernández-Medina).

0920-5489/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.csi.2010.01.006

4.9. Mead and Stehney "Security Quality Requirements Engineering (SQUARE) Methodology" [35], Mead and Hough "Security Requirements Engineering for Software Systems: Case Studies in Support of Software Engineering Education" [36] and Abu-Nimeh et al. "Integrating Privacy Requirements into Security Requirements Engineering" [37] . . . 158
4.10. Mellado et al. "A Common Criteria Based Security Requirements Engineering Process for the Development of Secure Information Systems" [38], "Towards security requirements management for software product lines: a security domain requirements engineering process" [39] . . 158
4.11. Moffett and Nuseibeh "A Framework for Security Requirements Engineering" [13] and Haley et al. "Security Requirements Engineering: A Framework for Representation and Analysis" [40] . . . 158
4.12. Morimoto, et al. "A Security Requirement Management Database Based on ISO/IEC 15408" [41] and Horie et al. "ISEDS: An Information Security Engineering Database System Based on ISO Standards" [42]. . . . 158
4.13. Myagmar et al. "Threat Modeling as a Basis for Security Requirements" [43] . . . 158
4.14. Peeters "Agile Security Requirements Engineering" [44] . . . 158
4.15. Popp et al. "Security-Critical System Development with Extended Use Cases" [45], Jürjens "UMLsec: extending UML for secure systems development" [46] and Best et al. "Model-Based Security Engineering of Distributed Information Systems Using UMLSec" [67] . . . 159
4.16. Shin and Gomaa "Software requirements and architecture modelling for evolving non-secure applications into secure applications" [48] . 159
4.17. Sindre and Opdahl "Eliciting security requirements with misuse cases" [49], Sindre et al. "A Reuse-Based Approach to Determining Security Requirements" [50], Opdahl and Sindre "Experimental comparison of attack trees and misuse cases for security threat identification" [51], Stalhane and Sindre "Safety Hazard Identification by Misuse Cases: Experimental Comparison of Text and Diagrams" [52], Whittle et al "Executable Misuse Cases for Modeling Security Concerns" [68], and Braz et al. "Eliciting Security Requirements through Misuse Activities" [69] . . . 159
4.18. Toval et al. "Requirements Reuse for Improving Information Systems Security: A Practitioner's Approach" [53], Martínez et al. "An Audit Method of Personal Data Based on Requirements Engineering" [54], Nicolás et al. "A Collaborative Learning Experience in Modelling the Requirements of Teleoperated Systems for Ship Hull Maintenance" [55] . . . 159
4.19. Tsoumas and Gritzalis. "Towards an Ontology-based Security Management" [57] and Tsoumas et al. "Security-by-Ontology: A Knowledge-Centric Approach" [58] . . . 159
4.20. Viega "Building security requirements with CLASP" [5] . . . 160
4.21. Yu "Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering" [59], Yu et al. "A Social Ontology for Integrating Security and Software Engineering" [60] and Yu "Social Modeling and i*" [61] . . . 160
4.22. Zuccato "Holistic security requirement engineering for electronic commerce" [62] and "Holistic security management framework applied in electronic commerce" [63]. Zuccato et al. "Security Requirements Engineering at a Telecom Provider" [64] . . . 160
5. Results and discussion . . . 160
6. Conclusions . . . 163
Acknowledgments . . . 163
References . . . 164

## 1. Introduction

The proliferation of connectivity of Information Systems (IS) and the increasing complexity of applications and services, signify that there is a correspondingly greater chance of suffering security breaches [1]. Present-day information systems are vulnerable to a host of threats and cyber-attackers such as malicious hackers, code writers, cyber-terrorists, etc. [2]. In addition, owing to the heavy dependence of computer network-based applications on various software and software controlled systems, the consequences of a security breach in these applications may range from extensive financial losses to dangers to human life. The threat of technology-enabled crime has given rise to a growing demand for the creation of new response strategies [2]. Software security has therefore become an essential issue [3] and a fair amount of additional security expertise is needed to meet non-functional security requirements [4].

However, security is rarely at the forefront of stakeholders concerns, except perhaps to comply with basic standards or legal requirements. Hence, work in requirements has primarily focused on eliciting and representing concrete business requirements [5], whilst requirements engineers often fail to pay sufficient attention to security concerns. The biggest problem, however, is that in the majority of software projects security is dealt with when the system has already been designed and put into operation. In addition to this, the actual security requirements themselves are often not well understood. This being so, even when there is an attempt to define security requirements, many developers tend to describe design solutions in terms of protection mechanisms, rather than making declarative propositions with regard to the level of protection required [6]. As a result, and perhaps for these reasons, although security requirements engineering has recently attracted increasing attention, it has lacked a systematic review which would supply researchers with a summary of all the existing information about security requirements in a thorough and unbiased manner, thus providing a context in which to operate.

Software Security Engineering, which is a practice through which to address software security issues in a systematic manner, is known to be a very important part of the software development process for the achievement of secure software systems. Nevertheless, within this discipline we believe in the particular importance of Security Requirements Engineering, which provides techniques, methods and norms for tackling this task during the early stages of the IS development cycle, since the building of security into the early stages of the development process is cost-effective and also brings about more robust designs [7]. It should involve the use of repeatable and systematic procedures in an effort to ensure that the set of requirements obtained is complete, consistent, easy to understand and analyzable by the different actors involved in the development of the system [8]. A good requirements specification document should include both functional requirements (related to the services that the software or system should provide), and non-functional requirements (related to what are known as features of quality, performance, portability, security, etc). In our contemporary Information Society, depending as it does on a huge number of software systems which play a critical role, it is absolutely vital to ensure that IS are safe right from the very beginning [9].

During the last few years, a number of papers have focused on security requirements, some of which have carried out reviews on this issue. However, most of these reviews consist of only one section in the paper/article and there are very few papers in which a review of security requirements is the core. After performing preliminary searches aimed at both identifying existing systematic reviews and assessing the volume of potentially relevant studies, we can highlight several works in which a summary of security requirements related issues is carried out, such as [3,10–13]. However, none of them perform a review focused on security requirements engineering in a systematic manner, that is, none of them perform a systematic review