To Comply Software and IT System Development with Related Laws

Fatemeh Zarrabi
Supervising team: Haris Mouratidis, David Preston, Shareeful Islam
School of Computing, Information Technology and Engineering, Uel University, United Kingdom
Email:shadi.zarrabi@uel.ac.uk

Abstract. Accretion procedure of crimes and security breaches against the privacy of individual's information and their maintenance information systems has cost huge amount of financial and other resources loose. Consequently governments take serious actions toward approving protective legislation against cyber crimes and it will be duty of software developers to adopt policies and measures to ensure that their designed systems are compatible with existing laws and their amendments. Since information technology and legislation are two quite distinct sciences, existence of a mechanism to do this adjustment and satisfy security and legal requirements of a designing software system is very essential. This paper is representing a framework that will help IT professionals to extract security requirements from relevant rules and use them in design of a system which is in accordance with those rules. It is giving brief discussion of the framework's methodology and design of a simulating computer-aided system of this framework. It also reports the research progress and new discovered conclusions.

Keywords: SDLC (System Development Life Cycle), goal modelling Language, goal-oriented software engineering, goal-agent modelling Language, Agent oriented software engineering, Hohfeld Law Modelling Language, Security Touch-points, Secure Tropos, SecTro

1. PROBLEM STATEMENT

The increasing demand of software systems to process and manage sensitive information and high interest of criminal activities for illegal access to these information [10], is the reason of high rate of approved laws and regulations against cyber crimes in these days. But the problem is that passing laws without implementing them in practical software systems, does not solve the origin of the problem and it is need to consist software systems with relevant rules and laws.

Since matching software systems with relevant laws is a new field of application and research, most of software developers have experienced same challenges regarding this evolution. The problem was that they had to review whole software development processes after the product release to match it against security and legal requirements. This process would lead to considerable costs of financial, effort and time resources. In reality this attempt had been an extra effort as response to legal authorities' pressure for compliance. For instance it has been estimated that in the Healthcare domain, organizations have spent $17.6 billion over a number of years to align their systems and procedures with a single law, the Health Insurance Portability and Accountability Act (HIPAA), introduced in 19961[8] . In the Business domain, it was estimated that organizations spent $5.8 billion in one year alone (2005) to ensure compliance of their reporting and risk management procedures with the Sarbanes-Oxley Act [8].

Therefore, need for synchronize analysis of legal requirements and other requirements of software systems is an essential and cost effective strategy of development. Thought there are several contributions in the area of software security requirements extraction and even in legal requirements extraction, still more can be done to have a specified framework which easily enables extraction of security and legal requirement from legal original texts and also aligns security and legal requirements in whole stage of software system development. However, eliciting, understanding, and satisfying system requirements that comply with relevant legislation is a challenging task because the concepts and terminology used for requirements engineering are mostly different to those used in the legal domain and there is a lack of appropriate modelling languages and techniques to support such activities. Also such a task is an extremely complex construction and time consuming for non-professionals in the field of law such as system developers. Therefore, research should be devoted to the development of techniques that systematically and automatically extract and manage requirements from laws and regulations in order to support requirements compliance to such laws and regulations. Having such a requirement extraction technique, the other challenge will be usage of it in different legal authorities region and if this system is flexible to the changes in different areas.

To categorize main mentioned problems, numbers of research questions are summarized here:

- How can we design reliable systems that are compatible with the relevant laws?
- Having a reference framework for the adaptation of software system to related laws, what components should be considered?
- How is it possible to design the automation of this framework which is also flexible to the changes of laws?

2. RELATED WORKS

Chung in [1] has introduced a framework called NFR (Non-Functional Requirements). In this research he has tried to extract all non-functional requirements of a system. His employed extraction technique is a goal and process oriented methodology in requirement engineering. Researchers in [2] have used GORE (Goal Oriented Requirement Engineering) Methodology to demise legal documents to threats and risks. In [3] Mead has proposed a method called SQUARE to elicit and document security requirements from relevant regulation. The work also agrees that security should be considered from first cycles of development. It almost refers to NIST (National Institute of Standard and Technology).
Ghanavati et al. [4] use User Requirement Notation based on Goal-oriented Requirement Language for a requirement management framework by modeling hospital business process and privacy legislation in terms of goals, tasks, actors, and responsibilities.
[5] By Mouratidis has tried to add security concepts to an existed requirement engineering methodology. Secure TROPOS was introduced by him which was an extension of the TROPOS project as an AOSE (Agent Oriented Software Engineering) methodology. Secure TROPOS has introduced security constraints concept into different stages of SDLC.Islam [8] extended it with security attack scenarios, where possible attackers, their attacks, and system resources are modeled. Islam [6] also proposed a goal-based software development risk management model (GSRM) to assess and manage risks from the RE phase.
The work which is represented in this paper is using an AOSE (Agent Oriented Software Engineering) technique to extract security and legal requirements from legal texts. Reason of choosing this technique is the close language and system process methodology of it to the concepts and terms used in legal texts since it is using concepts such as actors, their dependencies, goals and tasks.
In Goal Oriented Software engineering methodologies, functional characteristics are modeled as goals and these high level goals can be successively decomposed into lower-level goals and soft-goals and ultimately operationalized as actors, tasks and resources. In contrast, its process direction is in reverse of what is aimed here to extract requirements from laws, while this issue does not apply about AOSE [7].
This research tries to extend works by Mouratidis and Islam, to consider greater range of rules from financial environment, review its components and exercise other possible risk analysis techniques and finally design the framework and its computer-aided model in a way that it can align security and legal requirement extraction process. It also studies the feasibility of the framework's capability to be flexible to the changes of law.

3. PROPOSED APPROACH

To achieve the goal of this project which is to comply software system development with relevant laws, numbers of sequential processes are employed which each process's final output is the input to the next stage. A brief discussion of each process is provided here to give an overview of the framework of our project.

Activity 1: Methodology election. The first stage of the work had been to study different requirement engineering techniques to select the most suitable candidates for our framework. Since we are going to have a combination of requirement engineering and risk analysis techniques both in this framework, selection is from both areas.
Real world experiences and studies of Smart professional believe that security actions should be implemented in specific points of SDLC and failure of this will cause to security breaches. These points are vulgarly called as security touch-points[11]. The following figure is an image of our proposed framework considering software Development early stages (Requirement Analysis, Architectural Design and Late Design) and the security touch-point's locations in it.

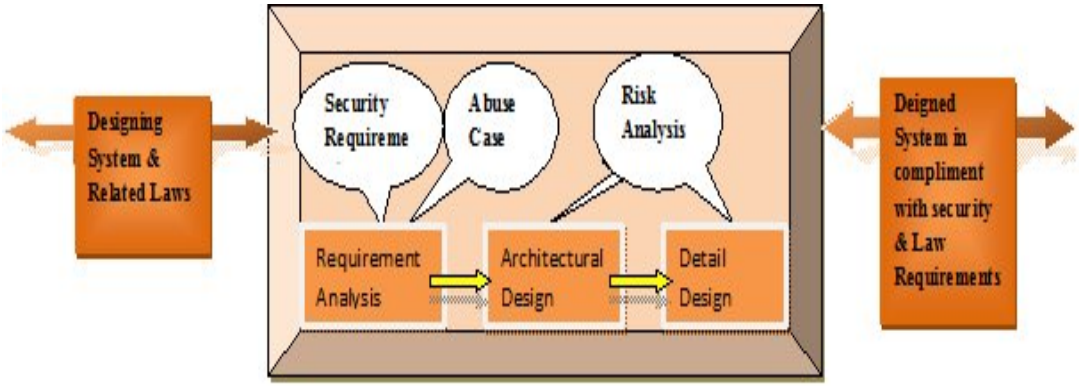

Fig1. Overview of the framework

Based on this model, aim of this research is to implement a methodology which almost simulates the life cycle of system development with the attached security touch-points to that. To do this it is essential to have a system development methodology which is based on the above step-by-step of SDLC and then implement and influence security actions and align them with legal requirements. This methodology will be used as the base of the study.

A candidate for such a system was almost from goal-oriented and goal modelling language methodologies since they have implemented a language very close to the language of law using terms such as actor, agent, dependency, soft goal, hard goal and etc. these concepts clearly can be mapped to law concepts such as parties and their relationships like their belonged rights and duties to each other.

Considering mentioned points, followings are proposed components of the desired framework:

Case study selection: To validate the study with a real word scenario, existence of a case study is essential in the research. Regarding the high rate of security attacks against financial and electronic payment systems and validity of a resource called as Globalplatform as an international specification organization for smart cards, this reference will be used as the case study of the research.

Activity 2: Case Study selection. To experience the validity of framework, a real world case study is chosen from financial area and will be used it in step by step process of our framework. Consequently all chosen laws are related to this case study and the transaction and relations between its stakeholders.

Activity 3: Model Regulations. Its aim is to elicit legal constraints from relevant legal texts and derive the legal rights from them. The resources which are going to be used as the reference of laws and regulation will be mostly Common Criteria and Information Security Standards, Information Security Laws from EU (European Union), Information Society Legislation, and finally any place relevant from National Legislations. Of course not all laws of this resources but only the one related to the chosen case study will be examined here. The work will use Hohfeld as a fundamental legal concepts classification model to analyze and model the regulations to extract possible legal rights from regulations. Reason of using Hohfeld is that this pattern helps to a precisely understanding of law [9]. Since legal texts are difficult to interpret by technical people from software system developer industry, Hohfeld will help simplifying of legal texts. The other reason is that laws are indeed expressed in terms of rights using concepts of claim, privilege, power, immunity and their correlatives like duty, no-right, liability and disability from regulations. Therefore using an international accepted pattern like Hohfeld which derive these fundamental concepts from laws is an advantage here.

Activity 4: Elicit Security Requirements from the modelled regulations. Secure Tropos is a system analyzer tool which has very early stages of SDLC. Here it will be used for security and legal requirement extraction, architectural and detail design of the case study and as a component of the framework. Secure Tropos is an extension of Tropos, which uses the concepts of actor (entity that has strategic goals and intentionality), goal (an actor's strategic interest), soft-goal (goal without clear

criteria whether it is satisfied or not), task (it represents the way of doing something), resource (it represents a physical or informational entity, without intentionality) and social dependencies (indicate that one actor depends on another in order to attain some goals, execute some tasks, or deliver a resource). Secure Tropos has redefined these concepts by adding security concepts and also it has the capacity for more concepts extension. Concepts of actors, their task and dependencies can clearly simulate legal concepts of relevant law parties, their duty and rights to each other and also will align these relationships with relevant security constraints.

Activity 5: Validate requirements with risk analysis techniques. To make sure if the extracted security requirements meet real world risks and threats, we will evaluate them by risk analysis methods like misuse or abuse cases. The place where in SDLC we use risk analysis is based on touch points order as discussed before. Research of more risk analysis methods may lead the usage of different risk analysis techniques rather than which are provided by touch point model.

Activity 6: Extension of methodology with new discovered law and security concepts. Final part of our work is to extent Secure Tropos with discovered security and legal concepts. The extended version of Secure Tropos can be used as an automated service by system developers to synchronize security and legal requirement extraction of a designing software system.

4. RESEARCH PROGRESS:

Based on the proposed research methodology and approach, GlobalPlatform is chosen as the case study of research. GlobalPlatform is a cross industry, not-for-profit association which identifies, develops and publishes specifications which facilitate the secure and interoperable deployment and management of multiple embedded applications on secure chip technology. Its proven technical specifications are regarded as *the* international industry standard for building a trusted end-to-end solution which serves multiple actors and supports several business models.

Technical documents provided by GlobalPlatform consist of specifications for cards, devices and systems of smart cards financial payment. Since aim of our case study is to design the financial payment with smart cards and not the design of the smart card or other devices, the only considerable references of GlobalPlatform will be system technical papers. To clarity how the whole system works, it will be beneficial to study other technical documents about cards and other devices.

The most relevant system technical documents to my proposed case study are listed below:

Smart Card Management System (SCMS) Functional Requirements v4.0: this document specifies and introduces the functional requirements of the proposed system. It helps to understand main goals of the system and to design the system and extract non-functional requirements.

Messaging Specification v1.0: this is a very good resource to understand and distinguish different parties (actors and agents) of the system from design of smart card and applications to financial transactions after the design process. It also discusses all messages and transactions between these parties and their relations to each other. This document is being used as the main resource for the system design.

Key Management Requirements and Systems Functional Requirements Specification v1.0: messages, applications and most of data which are transferred between different parties have to be encrypted for security reasons. Functions related to transaction of these encrypted data and their encryption keys are mentioned in this document.

Guide to Common Personalization v1.0: personalization is about loading the smart cards with some unique applications and information. Instructions related to this process are discussed in this document.

Load and Interface Specification v1.0: application loading functions are explained here in details.

Based on GlobalPlatform, smartcard design processes consists of two major categories: pre-issuance and after-issuance. Pre-issuance includes processes to issue the card and deals with actors like card manufacturer or Application developer. After-issuance deals with after card processes such as updating the card application and relevant actors are such as issuer and card holder. We are not going to design the functions of pre-issuance but just after-issuance processes. To understand how the whole system works and how is its actor's relationship, both pre and after issuance processes are considered only in high level design of the system. Secure Tropos which is a goal-agent oriented system analysis technology is being used as one of the framework's component to extract security and legal requirements of system and to plan architectural and detail design of it. It represents the system, its actors and actor's relationship by using of graphical symbols and diagrams. Images IV.I and IV.II are

samples of these diagrams designed by Secure Tropos tool that represent relevant actors of the proposed case study and relationships and transactions between them.

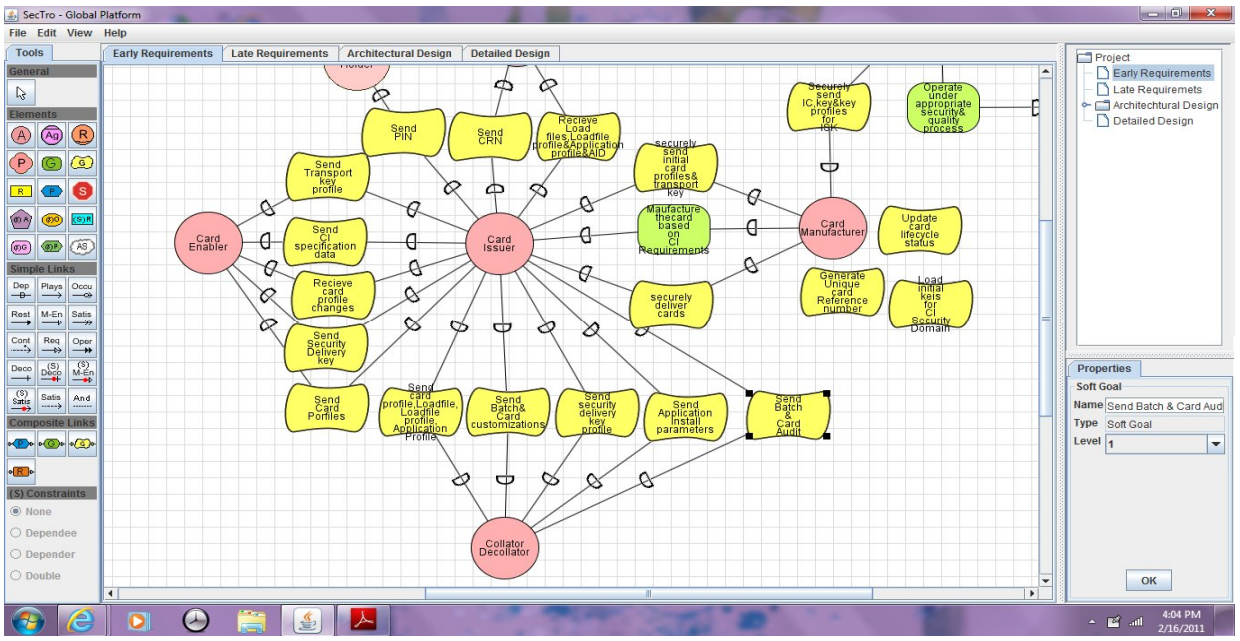

Fig2. GlobalPlatform Case Study Early Requirements Design by SecTro

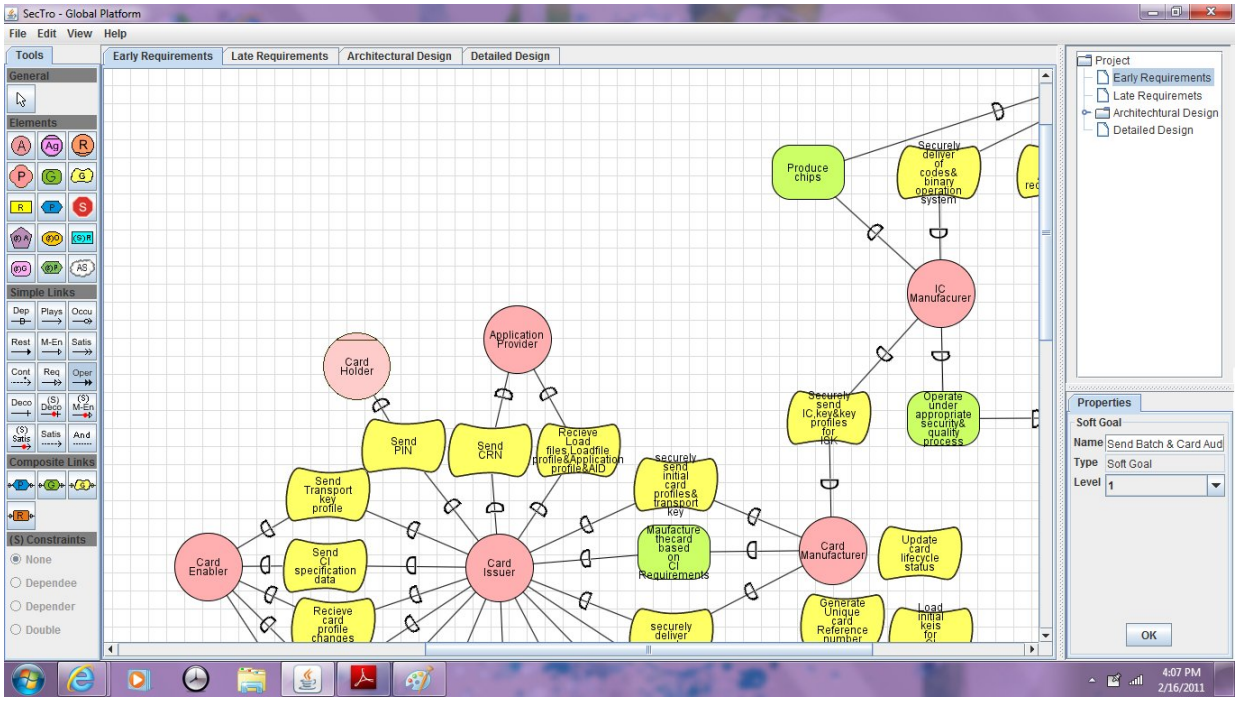

Fig3. GlobalPlatform Case Study, Early Requirements Design by SecTro

Later, security requirements of the various professionals involved with the system will be identified with the aid of security related concepts such as security constraints and secure goals. This enables system developers to understand the security requirements of the system by analyzing the security related concepts imposed to the various stakeholders/ users of the system. Then, these requirements can be transformed to security related functionalities that the system has to satisfy. Same process can be performed to extract legal requirements of the system by using of legal concepts.

To gain such legal concepts, relevant laws and administrative controls related to this case study should be studies and modelled. Since the most security requirement relevant to this case study and its actor's transaction is encryption of transferred data between them, the most related law will be encryption law.

Regarding threats and attacks that may happen against data stored or transferred, laws such as computer misuse ACT should be included as well. A list of most relevant laws to this case study is mentioned below:

- Law of Confidence
- Electronic Communication Act 2000
- Data Protect Act 1998
- Criminal IT law
- Computer Misuse Act 1990

Other standards and administrative controls resources are provided also by Common Criteria and even also by GlobalPlatform. Since the mentioned laws mostly are giving very high level and non-technical instructions and also since some laws like Electronic Communication almost deals with related stakeholder's certificate approval, more advantage will be to study other resources of administrative controls. Same circumstances come with other laws. Therefore there should not be worries regarding the huge amount of law's contents and the time limit of this research as not all parts of these laws are related to technical issues of information security.

Some example will help to the better understanding of the whole requirement extraction process in this project.

Based on above diagrams in image IV-II, card Holder depends on Card Issuer to receive PIN code from him. For security reasons, this information should be encrypted to be reached from card Issuer to Card Holder. This is a place where encryption legal instruction should be enforced and legal concepts like duty of confidentially for both sender and receiver should be introduced. Same time attacks and security risks against the transferred data should be analyzed. This process will align security and legal requirement in a same point of system design.

Other actors dependency with related security and legal requirements will be analyzed in same method as discussed.

5. CONCLUTION

As a response to necessity of legal compliance of IT systems, system developers are asked to consider law and legal issues in their system development. Since most of these laws deal with security aspects of information systems, legal and security requirements can be aligned together. Since smart's experiences have proved more accuracy and benefit of considering security from early stages of SDLC, it is necessary to have an automated system that can extract and align security and legal requirements and implement them into every and each stages of SDLC.

This paper advances the current state of the art by contributing the foundations of a framework that aligns security and privacy requirements with relevant legislation. The framework provides numbers of benefits. First it has the capability to translate the high level language of laws to a more understandable language for non-professional. Second benefit is its ability to model and extract security requirements from laws. The other advantage is its ability to align and synchronize legal and security requirement extraction from early stages of software system design process. Same time this framework is giving knowledge of legal and risk analysis resources (relevant laws, administrative controls and attack scenarios) to system developers.

Based on current stage of the research and studying relevant laws to the case study, new concepts of law has been discovered. Some examples of these concepts can be mentioned as "Duty of Confidence" and "Duty of information Disclosing". The first one applies to the system's actors when based on law, they are responsible to take action toward confidence transaction or maintenance of information (encryption). Second one applies when actors are required to disclose their stored or encrypted information to legal authorities for the aim of any investigation. Same concepts can be defined by using of other right's terminologies in the modelled law such as "Power of Access" or "Disability of Access" and other concepts. All these concepts can be used to extend current software engineering methodologies to align security and legal requirement extraction of a system.

6. REFERENCES

[1] L. Chung, B.A. Nixon, E. Yu and J. Mylopoulos, Non-Functional Requirements in Software Engineering, Kluwer Academic Publishers, 1999

[2] R. Darimont, M. Lemoine, Goal-oriented Analysis of Regulations, Regulations Modelling and their Validation and Verification

[3] N.R. Mead, Identifying Security Requirements Using the Security Quality Requirements Engineering (SQUARE) Method, in Integrating Security and Software Engineering, pp. 44-69, Idea Publishing Group, 2006

[4] S. Ghanavati, D. Amyot, and L. Peyton. Towards a framework for tracking legal compliance in healthcare. In J. Krogstie, A. L. Opdahl, and G. Sindre, editors, 19th International Conference on Advanced Information Systems Engineering (CAiSE'07), pp.218–232. Springer, 2007.

[5] H. Mouratidis. A Security Oriented Approach in the Development of Multiagent Systems: Applied to the Management of the Health and Social Care Needs of Older People in England. PhD thesis, University of Sheffield, U.K., 2004

[6] S. Islam, J. Jürjens, Incorporating Security Requirements from Legal Regulations into UMLsec model, Modelling Security Workshop (MODSEC08), In Association with MODELS '08, Toulouse, France, September, 2008.

[7] A. Herrmann, D. Kerkow and J. Doerr, "Exploring the Characteristics of NFR Methods – a Dialogue about two Approaches," REFSQ - Workshop on Requirements Engineering for Software Quality (2007), Foundations of Software Quality, 2007.

[8] 2Online News published in DMReview.com, November 15, 2004

[9] W. N. Hohfeld, fundamental Legal Conceptions as Applied in Judicial Reasoning, Yale Law of Journal 23(1), 1913.

[10] http://www.crime-research.org/news/21.02.2011/3861/

[11] G. McGraw, Software Security: Building Security In, Addison-Wesley Professional, pp.448, 2006.