Secure Information Systems Engineering: A Manifesto

Haralambos Mouratidis

Innovative Informatics, School of Computing and Technology, University of East London.

Abstract

In this paper we lay down the agenda for a discipline that is meant to promote research on increasing the development of secure information systems. In particular, we introduce areas related to the development of secure information systems; we identify limitations of existing approaches and the barriers that currently limit research, and we discuss the characteristics for an engineering discipline for the development of secure information systems, its principles and the challenges that must be addressed.

Keywords: information systems engineering, security engineering, secure information systems development, security requirements, integration of security and information systems engineering.

1 Introduction

We are living in a world where most of the information systems must be secure or they will not be used. Consider, for instance, the implications of a bank or a health care information system without provisions for security. It is therefore of paramount importance to fully understand the characteristics, principles and challenges that underlie the development of secure information systems. It is only then that we will be able to develop secure information systems. In gaining an in depth understanding of developing secure information systems, security should be considered, along with its related concepts such as trust and safety, within the whole context of information systems development and not in isolation. Moreover, various factors that might affect the security of an information system but are not limited to technical issues, such as for example the human factor, should be considered.

In fact, securing information systems raises a set of intertwined issues in the areas of security engineering and information systems engineering. However, information systems engineering and security engineering research communities traditionally work independently. As a result of this situation, security is usually considered after the analysis, design and implementation of the system has been completed. Security mechanisms are enforced into the system without considering the overall design and this usually results in problematic systems and security vulnerabilities (Stallings, 1999; Anderson, 2001).

We believe that the existence of secure information systems cannot be achieved just by employing formal models, methodologies and security mechanisms (although these are useful) during their development neither by ad-hoc approaches to solve the various problems involved in securing information systems. What is really needed is an engineering discipline which will form the basis to understand in depth the security issues involved in the development of information systems; provide the appropriate knowledge to assist information systems engineers and security engineers in developing secure information systems and also educate system users on issues related to the security of information systems. The main aim of this paper is to propose such an engineering discipline for secure information systems development, which we call Secure Information Systems Engineering.

The rest of the paper is structured as follows. Section 2 provides a brief introduction on security of information systems and section 3 discusses the motivation for secure information systems engineering. Section 4 lays out the manifesto for secure information systems engineering and section 5 concludes the paper.

2 Security of information systems

Physical security systems have been around for many thousands of years, ranging from castle fencing, to window bars and door locks. Computer security, on the other hand, although newer in comparison with physical security is definitely not a new topic since its history starts in the 1960s (Saltzer, 1975). Nevertheless, it was until the advent of distributed systems and computer networks that security of information systems has become an issue of huge concern.

According to Anderson (2001), "*security engineering is about building systems to remain dependable in the face of malice, error or mischance*". Therefore, security of computer based information systems is concerned with methods providing cost effective and operationally effective protection of information systems from undesirable events (Lane, 1985). Security is usually defined in terms of the existence of any of the following properties:

- *Confidentiality*: The property of guaranteeing information is only accessible to authorised entities and inaccessible to others.
- *Authentication*: The property of proving the identity of an entity.
- *Integrity*: The property of assuring that the information remains unmodified from source entity to destination entity.
- *Access Control*: The property of identifying the access rights an entity has over system resources.

- *Non repudiation*: The property of confirming the involvement of an entity in certain communication.
- *Availability*: The property of guaranteeing the accessibility and usability of information and resources to authorised entities.

Failure of any of the above-mentioned security properties might lead to many dangers ranging from financial losses to sensitive personal information losses. The existence of the above security properties within a system is defined in terms of the security policy. A *security policy* can be defined as "*the set of rules that state which actions are permitted and which actions are prohibited*" (Gollmann, 2001). A security policy determines the limits of acceptable behaviour and what the response to violations should be; and it might define possible mechanisms, widely known as *security mechanisms*, designed to detect, prevent or recover from a security attack. A *security attack* is defined (Stallings, 1999) as an action that compromises the secure information owned by an organisation or an individual.

It is well known that perfect security is very hard to achieve and usually the goal is to provide an acceptable security level, usually by trading security requirements with other functional and non-functional requirements of the system-to-be. Due to the attention that the issue of securing information systems has received the last few years and due to the large increase on the number of emerging defence mechanisms deriving from the ongoing research advances, someone would expect that system developers are able to develop and deploy very secure information systems. Nevertheless, current surveys indicate that we are far even from developing acceptable secure information systems (CERT, 2003; DTI, 2004).

One of the reasons is that, so far, security is mainly considered a technical challenge. However, it is has become apparent that a technical only approach in the development of secure information systems will not produce the expected results, since security is a multidimensional issue that cannot be considered in isolation. All information systems are ultimately embedded in some human social environment, and therefore the effectiveness of the system depends very much on the forces in that environment (Yu, 2006). Especially, with the advances on information systems and the transition towards open and autonomous systems, issues such as sociality, trust, privacy and delegation of responsibilities are closely related to the security of information systems. This argument is also supported by recent research, which has shown that the human factor has a significant impact on security. For example, one of the main threats to medical private records is social engineering. Social engineering is a non-technical kind of intrusion that relies on human interaction and involves tricking other people (doctors, or nurses in the case of medical records) to break normal security procedures.

3 Motivation for secure information systems engineering

There are various reasons that motivate the establishment of the secure information systems engineering discipline. In this section we identify and discuss the five most important of them, and we explain how these affect the development of secure information systems by presenting real-life scenarios.

3.1 Independent solutions

Securing information systems raises a set of intertwined issues in the areas of security engineering and information systems engineering. However, information systems engineering and security engineering research communities traditionally work independently. On one hand, information systems engineering techniques and methodologies do not consider security as an important issue, although they have integrated concepts such as reliability and performance, and they usually fail to provide precise enough semantics to support the analysis and design of security requirements and properties (Crook, 2003; Mouratidis, 2004). On the other hand, security engineering research has mainly produced formal and theoretical methods, which are difficult to understand by non security experts and which, apart from security, they only consider limited aspects of the system.

3.2 Problems in current state of the art

As indicated by current research (Mouratidis, 2006) information systems engineering methodologies do not create a security control environment early in the development process and modelling languages fail to include specialised handling of security requirements. However, there is a large number of works, which mostly have been developed the last few years.

Initial work from the information systems engineering community produced a number of methods and processes for reasoning about non-functional requirements, including security. Chung (1995) proposed the Non-Functional Requirements (NFR) framework to represent security requirements as potentially conflicting or harmonious goals and using them during the development of information systems. From the security engineering community, Schneier (2000) proposed attack trees as a useful way to identify and organise different attacks in an information system whereas Viega and McGraw (2001) proposed ten (10) principles for building secure software. More recently, Anton et al. (2004), proposed a set of general taxonomies for security and privacy, to be used as a general knowledge repository for a (security) goal refinement process.

The pattern approach has been proposed by a number of researchers to assist security novices to act as security experts. Schumacher and Roedig (2001) proposed a set of patterns, called security patterns, which contribute to the overall process of secure information systems engineering. Fernandez (2006) specified security models as object oriented patterns that can be used as guidelines for the development of secure information systems.

Although useful, these approaches lack the definition of a structured process for considering security. A well defined and structured process is of paramount importance when considering security during the development.

On the other hand, a number of researchers model security by taking into account the behaviour of potential attackers. Van Lamsweerde and Letier (2000) use the concept of security goals and anti-goals. Anti goals represent malicious obstacles set up by attackers to threaten the security goals of a system. In addition, Van Lamsweerde (2004) defines also the notion of anti-models, models that capture attackers, their goals and capabilities. Similarly, Crook et al. (2003) introduce the notion of anti-requirements to represent the requirements of malicious attackers. Anti-requirements are expressed in terms of the problem domain phenomena and are satisfied when the security threats imposed by the attacker are realised in any one instance of the problem. Lin et al. (2003), incorporate anti-requirements into abuse frames. The purpose of abuse frames is to represent security threats and to facilitate the analysis of the conditions in the system in which a security violation occurs. An important limitation of all these approaches is that security is considered as a vague goal to be satisfied whereas a precise description and enumeration of specific security properties is still missing.

Differently, another “school of thinking” indicates the development of methods to analyse and reason about security based on the relationships between actors (such as users, stakeholders and attackers) and the system. Liu et al. (2003) have presented work to identify security requirements, analysed as relationships amongst strategic actors, during the development of multiagent systems. Moreover, secure Tropos (Mouratidis, 2004) has been proposed to deal with the modelling and reasoning of security requirements and their transformation to design that satisfies them. Secure Tropos, is an extension of the Tropos methodology (Bresciani, 2004) and it is based on the concept of security constraint (Mouratidis, 2004; Mouratidis, 2005) to analyse the security requirements of an information system. To compliment the development process, security attack scenarios (Mouratidis, 2004b) and a security patterns language (Mouratidis, 2005c) have been developed. Giorgini at al. (2003) have introduced an enhancement of Tropos that is based on the clear separation of roles in a dependency relation between those offering a service (the merchant processing a credit card number), those requesting the service (the bank debiting the payment), and those owning the very same data (the cardholder). Moreover, Giorgini et al. (2004) have proposed a PKI / trust management requirements’ specification and analysis framework based on the clear separation of trust and delegation relationship.

Although a relationship based analysis is suitable for reasoning about security, an important limitation of these approaches is that either they focus on some development stages more than others (such as the secure Tropos approach) or they only guide the way security can be handled within a certain stage of the information systems development process (such as the work by Liu et al. and Giorgini et al).

Another direction of work is based on the extension of use cases and the Unified Modelling Language (UML). Initial work by McDermott and Fox

(1999) adapt use cases, which are called abuse cases, to capture and analyse security requirements. An abuse case is defined as a specification of a type of complete interaction between a system and one or more actors, where the results of the interaction are harmful to the system, one of the actors, or one of the stakeholders of the system. Similarly, Sindre and Opdahl (2005) define the concept of misuse case, the inverse of use case, which describes a function that the system should not allow. They also define the concept of mis-actor as someone who intentionally or accidentally initiates a misuse case and whom the system should not support in doing so. Alexander (2003) adds Threatens, Mitigates, Aggravates links to the use case diagram. Jurgens proposes UMLsec (Jurjens, 2004), an extension of the Unified Modelling Language (UML), to include the modelling of security related features, such as confidentiality and access control. Lodderstedt et al. (2002) also extend UML to model security. In their approach security is considered by analysing security related misuse cases.

An important limitation of all the use-case / UML related approaches, is that they do not support the modelling and analysis of security requirements at a social level but they treat security in system-oriented terms. In other words, they lack models that focus on high-level security requirements, meaning models that do not force the designer to immediately go down to security requirements.

On the other hand, a large amount of work has been devoted to security policies and the definition of security models. Various models[1] have been proposed based on mandatory access control (MAC), discretionary access control (DAC) and role base access control (RBCA). One of the first models was the Bell & Lapadula multilevel security model (Bell, 1976).

[1] An extensive presentation and discussion of these models are out of the scope of this chapter and this book.

Another well known model is the Chinese Wall model (Brewer, 1989), according to which data is organised into three different levels.

The definition of security ontology is also an important area of research within the security engineering community. Initial efforts to define a widely accepted security ontology resulted in what is known as the Orange Book (US Department of Defense Standard DOD 5200.58-STD). However, work towards this standard started in the late 1960s and it concluded in the late 1970s. Therefore important issues, raised by the introduction of the Internet and the usage of information systems to almost every aspect of our lives, are missing from the standard. More recently Kagal et al (2005) have developed an ontology expressed in DAML+OIL to represent security information, trust and policies in multiagent systems, whereas Undercoffer and Pinkston (2002) after analysing over 4000 computer vulnerabilities and the corresponding attack strategies employed to exploit them have produced an ontology for specifying a model of computer attacks.

Although important and useful in many situations, the above work has a number of important limitations with respect to the integration to information systems engineering practice. First of all, it mainly considers the later stages of the information systems development process. As argued before, it is important that security is considered from the early stages of the development process. Moreover, existing work is mainly focused on the technological aspects of security and it ignores, in general, the social dimension of security. It is important that security is considered within the social context and any social issues, such as trust and the involvement of humans, are taken into account (Mouratidis, 2006).

3.3 Custom solutions

In many cases the inclusion of security on a system is driven by existing custom solutions (security mechanisms) rather than the system’s real security requirements. Supporting the development of the security of the system on specific security mechanisms, as opposed to security requirements, prevents the consideration and choice of different and sometimes better solutions to satisfy the security requirements. For instance, imagine a system which requires identification and authentication. If the development of the system is based on some specific solutions to these requirements, such as username and password, then other solutions might be ignored, such as biometric identification and authentication, which in some cases could better fulfil the initial security requirements. Therefore, it is important that only the security requirements drive the development, as it happens with functional requirements, and not the well-known security solutions.

3.4 Lack of sharing existing knowledge

As indicated above, information systems and security engineering communities mainly work separately. This separation not only creates a void in the proposed solutions but it also results in restricted sharing of existing knowledge. Different research events organised by the two communities, different research publications and so on. Even widely used textbooks mostly concentrate in one part of the problem, either technical security issues or information systems engineering techniques, and they only contain, when they do, very limited information about the integration of the security and information systems engineering principles.

3.5 Lack of appropriate education

Professional training courses and university curriculum should help towards the solution of the above problem. However, they propagate it.

Information systems and security engineering training as well as curriculum development in universities follows the separation of the two research areas. As a result, information systems engineering principles are taught separated from security engineering issues and vie versa. This means that information systems engineers are not well educated regarding the security issues that might face during the development of information systems, and security engineers mostly are not familiar with current practices and issues surrounding information systems engineering.

3.6 Problematic Scenarios

The above problems affect the development of secure information systems as demonstrated by the following real-life scenarios:

- Requirements engineers do not usually receive appropriate training (Firesmith, 2003) in eliciting, analysing and specifying security requirements. As a result, they often confuse them with security mechanisms which are used to fulfil them. Therefore, they end up defining architectures and constraints rather than true security requirements (Firesmith,2003).
- Information systems engineers are faced with the development of secure information systems according to their security requirements. However, not all information systems practitioners are security specialists neither they fully understand mathematical security models (McDermott, 1999). An information systems engineer without the appropriate security knowledge and without information systems engineering practices that integrate security as part of the development process more likely will fail to develop the system according to its (security) requirements.
- Security engineers are often required to enhance the security of an existing system. However, current security models and

methodologies used by security engineers do not fully analyse nor reason the implication that the addition of security components will have on the existing functionalities of the system. Without appropriate processes and methodologies to guide them, most likely they will fail.

- Information systems engineers are required to test, during design, whether the system under development satisfies its security requirements. However, the lack of appropriate languages and automated techniques makes such task very difficult.

It has been widely argued (Anderson, 2001; Van Lamsweerde, 2004; Mouratidis, 2006) within the computing research community that a careful integration of security issues within information systems engineering processes will provide a solution to the above technical problems and a step towards the development of information systems with less security vulnerabilities. In particular, Devanbu and Stubblebine (2000) state on their roadmap on information systems engineering for security "*security concepts must inform every phase of software development, from requirements engineering to design, implementation, testing, and deployment*". Similarly Crook et al (2003) state "*our vision is that we will be able to model these [security] concepts and integrate them into the requirements engineering process*". In line with these statements, Mouratidis et al (2005) argued that security should be considered from the early stages of the development process and security requirements should be defined alongside with the system's requirements specification. Taking security into account alongside the functional requirements throughout the development stages helps to limit the cases of security/functional requirements conflict by avoiding them from the very beginning or by isolating them very early in the development process. To adequately consider security issues during the information systems development life cycle, security should be integrated within information systems engineering languages, methods, methodologies and processes.

We agree with the above views but we believe that ad-hoc approaches will not adequately resolve all of the above problems. Most often, and as it is demonstrated by the current state of the art, ad-hoc approaches tend to focus only on the technical problems, for instance the development of languages, methodologies, models and so on. However, such approaches, although they provide right steps towards the solution of the most difficult problem, do not provide any solutions towards the other problems, such as the lack of appropriate education and sharing of information.

To adequately resolve all the issues, we need a large scale effort, lead by the information systems and security engineering communities, to establish a discipline concerned with the development of secure information systems. The rest of the paper presents a manifesto of such a discipline, which we denote by the term secure information systems engineering.

4 Secure information systems engineering: A Manifesto

Although we provide a definition for secure information systems engineering, we do not consider it to be absolute, but rather we expect it to be revised from time to time to indicate the maturity and the advance of the discipline, as it is the case with most disciplines. We define Secure Information Systems Engineering as the engineering discipline concerned with the development of secure information systems. In particular, secure information systems engineering is concerned with the knowledge (theoretical and practical), principles, practices as well as the establishment of a research agenda regarding secure information systems development. The underlying aim of Secure Information Systems

Engineering is to improve the quality of information systems by reducing the number of security vulnerabilities that these systems demonstrate.

4.1 Characteristics

According to Liles et al. (1995), every discipline demonstrates six characteristics: (1) a focus of study; (2) a world view; (3) a set of reference disciplines used to establish the discipline; (4) principles and practices associated with the discipline; (5) an active research or theory development agenda; (6) and the deployment of education and promotion of professionalism.

4.1.1 A focus of study

Every discipline aims to address a unique fundamental question or the focus of study. Such a question must have enough substance to evolve into a classical field of study (Liles, 1995) and be independent of technological changes (Keen, 1980).

The fundamental question for secure information systems engineering can be formulated as "*how to develop secure information systems?"*. In answering such a question, many sub-questions need to be formulated and answered. For example, what we mean by "*secure information systems?*", "*what is good security?*", "*how do we define security requirements*?". Usually, different researchers and practitioners will answer differently such questions. However, it is imperative that common answers are established in such fundamental issues, in order to provide a well-founded base in which we will be able to base further research questions leading us closer to answer the fundamental question of the discipline.

4.1.2 A world view (or paradigm)

The way that the discipline views the world guides the development of the discipline through practice and research (Doheny, 1987). The viewpoint of a discipline needs to be complex and substantial enough to be divided into sub-disciplines or sub-fields (Keen, 1980).

We envisage the maturity of secure information systems engineering in such a degree that information systems developers will be able to model, construct, test, deploy and maintain secure information systems through well defined and structured processes and with the aid of appropriate modelling languages. In such a vision, development is made even easier with the aid of computer-aided tools that enable to accurately track the security solution to the initial system requirements and therefore validate it against the security goals of the organisation where the system is deployed.

Our vision is based on three main world view assumptions for secure information systems engineering: (1) the development of secure information systems is a complex issue which involves technical as well as social challenges; (2) Processes, models, methodologies and automated tools can be employed to address the technical challenges and to assist in the development of secure information systems; (3) Proper education of anyone involved in the development as well as in the usage of information systems is needed to support the outputs of research addressing the technical challenges and to compliment the social challenges.

The secure information systems engineering discipline can be divided into sub-disciplines such as security requirements engineering security modelling, secure information systems development, security policies / models / ontology and secure information systems engineering education.

4.1.3 A set of reference disciplines used to establish the discipline

Throughout history, new disciplines have emerged from the need to solve new problems that are not fully addressed by existing disciplines (Liles, 1995). For example, in previous section we have discussed a list of problems of existing research and practice which motivate the establishment of secure information systems engineering, and we have explained the reasons why existing disciplines such as information systems and security engineering have failed to fully address these problems.

However, disciplines do not exist in isolation but they are related to reference disciplines. Reference disciplines are existing bodies of knowledge that help establish the new discipline. Formally referencing disciplines recognizes the contributions of existing knowledge and provides a logical link to the new discipline. Without this linkage, researchers in existing disciplines may question the grounding theories of a new discipline and dismiss its importance (Liles, 1995).

Secure information systems engineering builds upon the knowledge, theories and methods of several existing disciplines including information systems engineering, security engineering, and social sciences. The development of such techniques should be based on research provided by the security engineering research community, such as attack testing, secure design principles and security ontologies, complimented by research provided by the information systems engineering community, such as requirements engineering techniques, information systems development methodologies and modelling languages, and testing. Moreover, theories coming from social sciences should also be taken into account to ensure that the human factor is appropriately considered.

4.1.4 Principles and practices associated with the discipline

Principles incorporate the world view and define the philosophical approach to solving problems. Practices are the methodologies, models, procedures, and theories used to apply the discipline’s knowledge base. Together, principles and practices form the foundation of a discipline and promote further ordered study. In an engineering discipline, the body of abstract knowledge is developed by logical analysis and scientific research. The principles and practices of engineering are embodied in systems of theory, abstraction, design, and implementation. It is the activities which occur inside these systems that differentiates the many engineering disciplines (Liles, 1995).

The proposed discipline has two main objectives: (1) the production of novel techniques, methods, processes and tools which will integrate security and information systems engineering principles; and (2) the education of information systems developers to use such techniques to analyse, design, implement, test and deploy secure information systems.

We argue that an engineering discipline for secure information systems should be based on the following principles: consider security from the early stages of the information system development; separation of concepts; ensure quality of security solution; consistency.

Although some of the above principles are not novel, and they are based on related information systems and/or security engineering principles, the point is that current approaches do not follow them.

4.1.5 An active research agenda

An active research agenda implies that hypotheses are being generated which address the fundamental question of the discipline. The agenda stands the test of time, with many researchers and practitioners in the discipline continually expanding the research that builds upon itself (Liles, 1995). We believe that the research agenda for secure information systems engineering should include the following challenge:

Challenge 1: *Unify efforts to integrate security and information systems engineering.*

Although the need for such unification has been recognised by various researchers, work on integrating security and information systems engineering is mainly carried out independently either by members of the security research community or by members of the information systems engineering community. It is important to unify the efforts on the two fields. Only then we will be able to precisely identify the technical as well as the social issues that surround the development of secure information systems and produce solutions that truly work.

Challenge 2: *Consider the social dimension of security*.

Security is mainly considered as a technical issue by information systems and security engineers alike. A mature solution that integrates security and information systems engineering should consider not only the technical dimension of security but also the social dimension. It is only when we consider both dimensions that we will be able to develop secure enough information systems.

Challenge 3: *Develop complete security ontology.*

The need for sound and complete security ontology is well recognized as an important issue for the development of widely accepted solutions on secure information systems engineering. Such ontology will provide a firm and well-understood foundation to support the development of appropriate methods, processes and methodologies.

At present, work on defining such ontology is carried out independently by the information systems engineering and security engineering research communities. This separation of work has resulted in an abstraction gap, which makes the integration and practical application of security issues on information systems engineering practices difficult. As an example, consider the term “security requirement”. Although this term is fundamental; so far, it is used and interpreted differently by various researchers and practitioners.

Challenge 4: *Define a suitable exemplar*.

Typically, in information systems engineering, various approaches will be demonstrated using case studies which are tailored to emphasise the key characteristics of the approach. However, such case studies often focus on specific problems. It is important, therefore, to define a suitable example problem (in information systems engineering community the term exemplar is widely used when referring to an example problem) which will emphasize the problems faced by the community and it will serve as a focal point for discussion and exchange of research ideas and results. In choosing such an exemplar various criteria should be considered. For instance, the exemplar should be broad enough to cover all the possible issues, technical or social, which are associated with the development of secure information systems. Moreover, it should be generic enough as well as rich and complex enough to test the limits of any proposed approach.

Challenge 5: *Evaluate the different information systems engineering paradigms with respect to their appropriateness to integrate security*

Various information systems engineering paradigms exist such as model-driven, aspect-oriented, and agent-oriented. All these treat information system development differently, using their own set of concepts and techniques. It is very important to identify the strengths and weaknesses of each of these paradigms when integrating security into the development process.

Challenge 6: *Development of new techniques, methods, processes that consider security as part of the information system development lifecycle.*

At present, most existing methodologies and models concentrate only on specific stages of the development process, such as security requirements engineering, or security design. It is vital, however, that security is considered throughout the development process and it is considered alongside the functional requirements and other non-functional requirements of the system-to-be. It is only then that we can consider security as part of the development process and not an isolated concept of the system. Therefore, it is important to develop new methods and techniques. These should support the formal (and simultaneous) modelling, reasoning and analysis of security and functional requirements, and the transformation of such (security and functional) requirements to a design that will satisfies them. Moreover, one of the main problems of considering security during the development stages of an information system is the lack of methods and techniques to trace the provided functionality to security requirements and also test the solution before the implementation of the system. Therefore, it is crucial to develop new methods and techniques to support traceability and validation of the proposed solution.

Challenge 7: *Tool support*.

Integrating security in the development process means adding extra activities in an already difficult task. Therefore, it is of paramount importance to produce tools to support the development process. A tool should not only support developers in modelling and reasoning about security (and functional requirements) during the analysis stage, but it should help to transform the requirements to design, check the consistency of the proposed solution and also validate the security functionalities of the proposed solution against the security requirements of the system.

Challenge 8: *Transfer of security knowledge*.

Many system developers do not always have a strong background in computer security and lack expertise in secure information system development. Nevertheless, in practice, they are asked to develop information systems that require security features. Secure information systems engineering methodologies should consider that issue and provide methods and processes that allow even developers with minimum security expertise to analyse and design a system with security in mind. At present, security patterns seem to provide a right step into this direction, as also argued in some of the chapters of this book. However, there is a need to enhanced current pattern languages and provide a better integration with information systems engineering processes and methods.

Challenge 9: *Transit research results to mainstream system development.* An important, long-term, challenge is the successful transfer of research knowledge and best practice on developing secure information systems to industry. To achieve this, there is a need to make secure information systems engineering practice widely known (research and industry), standardize them and provide an agreed set of techniques, models and methodologies. This will ensure trust in the proposed methods and industrial confidence.

4.1.6 Education and Professionalism

Education and professionalism are essential to the widespread recognition and deployment of a discipline. A discipline should be identifiable with a research community that sustains its own literature. The written record of knowledge and thought progression is valuable for future researchers and practitioners to reference when developing new theories and methodologies. Conferences and journals provide a forum for researchers and practitioners to exchange ideas, develop new knowledge and identify future lines of research. Separate curricula, professional societies, and journals advance professionalism and are necessary for a separate discipline (Maynard, 1971). Although some research events, such as SREIS (www.sreis.org) and ISSSE (www.jmu.edu/iiia/issse) have been successfully organised the last few years; there is a need to organise large scale events that will involve not only researchers from all the related research communities but also industrialists. There is also need to encourage a curriculum which incorporates the different aspects coming from the various interdisciplinary subjects in order to provide the required knowledge.

5 Conclusions

This paper argues about the need to form a discipline to promote secure information systems development. Such effort should bring together experience and techniques from information systems engineering, security engineering and social studies disciplines in a coherent and organised way. An attempt to define the aims, objectives, practices and the challenges of the proposed discipline is taking place. However, this is not an absolute attempt and the paper aims to motivate a large scale effort towards the development of the discipline, which will hopefully result into a more complete and detailed definition of the proposed discipline.

References

Alexander, I. (2003). Misuse Cases: Use cases with hostile intent. IEEE Software, 20, 58-66.

Anderson, R., (2001), Security Engineering: A Guide to Building Dependable Distributed Systems, Wiley Computer Publishing.

Anton, A.I., Earp, J.B., (2004) A requirements taxonomy for reducing web site privacy vulnerabilities, Requirements Engineering, 9(3):169-185, 2004.

Bell, D. E., LaPadula, L. J., (1976) Secure Computer Systems: Mathematical Foundations and Model. The Mitre Corporation

Bresciani, P., Giorgini, P., Giunchiglia, F., Mylopoulos, J., Perin, A., (2004). TROPOS: An Agent-Oriented Software Development Methodology, Journal of Autonomous Agents and Multi-Agent Systems. Kluwer Academic Publishers Volume 8, Issue 3, Pages 203 - 236.

Brewer, D.F.C., Nash M.J. (1989),The Chinese Wall Security Policy, Proceedings of the IEEE SYMPOSIUM ON RESEARCH IN SECURITY AND PRIVACY, pp.206-214, 1-3 May1989, Oakland, California. pp 206-14)

CERT Coordination Centre (2003), Annual Report, available at www.cert.org

Chung, L., and Nixon, B., (1995) Dealing with Non-Functional Requirements: Three Experimental Studies of a Process-Oriented Approach. In Proceedings of the 17th International Conference on Software Engineering, Seattle- USA.

Crook, R., Ince, D., Nuseibeh, B. (2003). Modelling Access Policies Using Roles in Requirements Engineering, Information and Software Technology. 45(14):979-991, Elsevier

Devanbu, P., Stubblebine, S. (2000). Software Engineering for Security: A roadmap. Proceedings of the 22nd International Conference on Software Engineering. Track on the Future of Software Engineering. Limerick –Ireland.

Doheny, M. O., Cook, C., Stopper, M., (1987), The Discipline of Nursing: an introduction, 2nd edition, Appleton & Lange, Norwalk, Connecticut.

DTI, Information Security Breaches Survey (2004), Technical Report, available at www.dti.gov.uk

Fernandez, E.B. (2004) A methodology for secure software design, Proceedings of the 2004 International Conference on Software Engineering Research and Practice (SERP'04), Las Vegas, NV, June 21-24, 2004.

Firesmith D.G., (2003). Engineering security requirements, Journal of Object Technology, Vol 2., No. 1, ETH Swiss Federal Institute of Technology

Giorgini, P., Massacci, F., Mylopoulos, J., (2003). Requirements Engineering meets Security: A Case Study on Modelling Secure Electronic Transactions by VISA and Mastercard, in Proceedings of the International Conference on Conceptual Modelling (ER), LNCS 2813, pp. 263-276, Springer-Verlag.

Giorgini, P., Massacci, F., Mylopoulos, J., Zannone, N., (2004). Filling the Gap between Requirements Engineering and Public Key/Trust Management Infrastructures. In Sokratis K. Katsikas, Stefanos Gritzalis, Javier Lopez (Eds.): Public Key Infrastructure, LNCS 3093 Springer, Proceedings of the First European PKIWorkshop: Research and Applications, EuroPKI 2004, Samos Island, Greece, June 25-26

Gollmann, D., (2001), Computer Security, John Willey and Sons.

Haley, C. B., Moffett, J. D., Laney, R., & Nuseibeh, B. (2005). Arguing Security: Validating Security Requirements Using Structured Argumentation. In Proceedings of the Third Symposium on Requirements Engineering for Information Security (SREIS'05) held in conjunction with the 13th International Requirements Engineering Conference (RE'05). Paris, France.

Jürjens, J., (2004). Secure System Development with UML. Springer-Verlag.

Kagal, L., Finin, T., (2005). Modeling Conversation Policies using Permissions and Obligations, in Developments in Agent Communication, Frank Dignum, Rogier van Eijk, Marc-Philippe Huget (Eds), (Post-proceedings of the AAMAS Workshop on Agent Communication, Springer-Verlag, LNCS), January, 2005.

Keen, Peter G. W., (1980), MIS Research: Reference Disciplines and Cumulative Tradition, *Proceedings of the First International Conference on Information Systems*, Philadelphia, Pennsylvania, December, pp.9-18.

Lane, V., P., (1985), Security of Computer Based Information Systems, Macmillan education Ltd.

Liles, D.H., Johnson, M.E., Meade, L.M., Underdown, D.R., (1995), Enterprise Engineering: A discipline?, Proceedings of the Society for Enterprise Engineering Conference, June.

Lin, L.C., Nuseibeh, B., Ince, D., Jackson, M., Moffett, J., (2003). Analysing Security Threats and Vulnerabilities Using Abuse Frames, Technical Report 2003/10, The Open University

Liu, L., Yu, E., Mylopoulos, J., (2003). Security and Privacy Requirements Analysis within a Social Setting, In Proceedings of the 11th International Requirements Engineering Conference, pp. 151-161, IEEE Press.

Lodderstedt, T., Basin, D., Doser, J., (2002). SecureUML: A UML-Based Modelling Language for Model-Driven Security, in Proceedings of the UML'02, LNCS 2460, pp. 426-441, Springer-Verlag.

Maynard, H.B., (1971). Industrial Engineering Handbook, 3rd Edition, McGraw Hill Book Co., New York

McDermott, J., Fox, C., (1999). Using Abuse Care Models for Security Requirements Analysis, Proceedings of the 15th Annual Computer Security Applications Conference.

McGraw, G., Viega, J., (2001), Building Secure Software: How to Avoid Security Problems the Right Way. Addison-Wesley.

Mouratidis, H. (2004). 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.

Mouratidis, H., Giorgini P., Manson, G., (2005). When Security meets Software Engineering: A case of modelling secure information systems, Information Systems, Vol. 30, Issue 8, pp. 609-629, Elsevier.

Mouratidis, H., Giorgini, P., Manson, G., (2004b). Using Security Attack Scenarios to Analyse Security During Information Systems Design, in the Proceedings of the International Conference on Enterprise Information Systems (ICEIS 2004),pp. 10-17, April, Porto-Portugal

Mouratidis, H., Weiss, M., Giorgini, P., (2005c). Security patterns meet agent oriented software engineering: a complementary solution for developing security information systems, Proceedings of the 24th International Conference on Conceptual Modelling (ER),Lecture Notes in Computer Science 3716, pp. 225-240, Springer-Verlag.

Mouratidis, H., Giorgini, P. (2006). Integrating Security and Software Engineering: Advances and Future Vision, IDEA Group Publishing, ISBN 1-59904-148-0.

Saltzer, J., Schroeder, M.D., (1975). The Protection of information in computer systems, In the Proceedings of the IEEE 63 (9), pp.1278-1308, September 1975.

Schneier, B., (2000). Secrets & Lies: Digital Security in a Networked World, John Wiley & Sons

Schumacher, M., Roedig, U., (2001). Security Engineering with Patterns, in the Proceedings of the 8th Conference on Pattern Languages for Programs (PLoP), Illinois – USA

Sindre, G., Opdahl, A.L., (2005). Eliciting security requirements with misuse cases, Requirements Engineering, 10(1):34-44

Stallings, W., (1999) Cryptography and Network Security: Principles and Practice, Second Edition, Prentice-Hall.

Undercoffer, J., Pinkston, J., (2002). Modelling Computer Attacks: A target-centric ontology for intrusion-detection, proceedings of the CADIP research symposium, available at: http://www.cs.umbc.edu/cadip/2002Symposium/

Van Lamsweerde, A., (2004). Elaborating Security Requirements by Construction of Intentional Anti-Models, Proceedings of the 26th International Conference on Software Engineering, Edinburgh, May, ACM-IEEE, pp. 148-157.

Van Lamsweerde, A., Letier, E., (2000). Handling Obstacles in Goal-Oriented Requirements Engineering, Transactions of Software Engineering, 26 (10): 978-1005
Viega, J., McGraw, G., (2001). Building Secure Software. Addison Wesley

Yu., E, Liu., L., Mylopoulos, J., (2006), A social ontology for integrating security and software engineering, in Integrating Security and Software Engineering: Advances and Future Vision, H. Mouratidis and P. Giorgini (editors), Idea Group Publishing, 2006