A Systematic Approach to Define the Domain of Information System Security Risk Management

Éric Dubois, Patrick Heymans, Nicolas Mayer, and Raimundas Matulevičius

Abstract Today, security concerns are at the heart of information systems, both at technological and organizational levels. With over 200 practitioner-oriented risk management methods and several academic security modelling frameworks available, a major challenge is to select the most suitable approach. Choice is made even more difficult by the absence of a real understanding of the security risk management domain and its ontology of related concepts. This chapter contributes to the emergence of such an ontology. It proposes and applies a rigorous approach to build an ontology, or domain model, of information system security risk management. The proposed domain model can then be used to compare, select or otherwise improve security risk management methods.

1 Introduction

During the last two decades, the impact of security concerns on the development and exploitation of Information Systems (IS) never ceased to grow, be it in public or private sectors. In this context, security Risk Management (RM) has become paramount because it helps companies identify and implement security requirements in a cost-effective manner. Indeed, security threats are so numerous that it is outright impossible to act on all of them, because (1) every technological security solution has a cost, and (2) companies have limited resources. Hence, companies need assurance that they adopt only solutions that will provide significant Return on Investment (ROI). This is done by comparing the cost of a solution with the risk of not using it, e.g., the cost of a business disruption due to a successful security attack. In this sense, security RM plays an important role in the alignment of a company's business strategy with its Information Technology (IT) strategy.

É. Dubois (✉)
Centre de Recherche Public Henri Tudor, 29, avenue John F. Kennedy,
L-1855 Luxembourg-Kirchberg, Luxembourg
e-mail: eric.dubois@tudor.lu

S. Nurcan et al. (eds.), *Intentional Perspectives on Information Systems Engineering*,
DOI 10.1007/978-3-642-12544-7_16, © Springer-Verlag Berlin Heidelberg 2010

Today there exist literally hundreds of IS Security RM (ISSRM) methods and standards targeted to professionals (see Sect. 3.2 for an overview). They mainly consist of process guidelines that help identify vulnerable assets, determine security objectives, and assess risks as well as define and implement security requirements to treat the risks. By using these methods one reduces the losses that might result from security problems. However, these methods generally offer very little modelling support. Instead, they usually resort to informal documentation in natural language and ad hoc diagrams. This means that powerful abstraction mechanisms, visualisations and automations offered by conceptual modelling techniques are underexploited.

On the contrary, the Requirements Engineering (RE) literature features a number of modelling languages specifically dedicated to security-sensitive contexts.Examples of such languages are Misuse Cases [51] and Abuse Cases [42], which extend Use Cases [7]; Abuse Frames [31–33] derive from Problem Frames [27]; Secure-Tropos [17, 46, 47] originates from Tropos [4] and *i** [57]; KAOS [30] was also extended [29] to deal with security aspects. The main benefit of these languages is to address security concerns in the early phases of IS development. This allows enforcing security *by construction*, which is more effective than doing it after the fact [48]. However, it turns out that these languages lack constructs to properly represent risk, e.g., vulnerable assets, their associated security risks and risk treatments (with the notable exception of [2] which supports a more general notion of risk). Hence, although these languages are useful in eliciting and modelling threats and countermeasures, they are still largely unable to address cost-effectiveness concerns in a satisfactory manner.

These observations could be used as arguments in favour of defining a new, more suitable modelling language. However, defining a completely new notation does not appear to us as a viable option for at least two reasons. Firstly, this would only further populate the already overcrowded jungle of modelling languages. Secondly, we aim at a smooth rather than radical transition from current practice. Existing languages address different complementary views (e.g., scenario-oriented view, goal-oriented view. . .), all potentially useful for RE. ISSRM actually crosscuts those views and should therefore be related to them. So, as long as this does not make the languages too complex, we rather plead in favour of improving existing languages with a better coverage of the ISSRM domain.

In this chapter, we do not go as far as proposing an extension to an existing language. Instead, we describe an intermediate step which is concerned with answering the following research question: *What are the concepts that should be present in a modelling language supporting ISSRM during the early stages of IS development?* In this, we follow a similar approach as those pioneers who designed IS modelling languages back in the eighties [49, 50]: first identify the key concepts of the subject domain, then design (or adapt) a language to support it.

The remainder of this chapter is structured as follows. Section 2 presents the research method that we have followed for answering this question. In Section 3 we introduce the basic definitions associated with security and risk management and present our survey of the literature. Section 4 proposes a synthesis of the surveyed

literature by means of a concept alignment table. The latter is further consolidated into a domain model for ISSRM presented in Sect. 5. Section 6 finishes our work with conclusions.

2 Research Method

Our overall research method (see Fig. 1) consists of four steps:

Step 1 – Concept alignment. We start by investigating the state of the art in ISSRM. Our goal is to identify the core concepts of the domain and harmonise the terminology. The main outcomes are:

- A *concept alignment table* that highlights the core concepts of the surveyed approaches and indicates synonymy or other semantic relationships when approaches use different terms;
- A *glossary* of the terms as found in the sources.

An excerpt of the table is shown in Table 1 (the complete table can be found in [38]). To obtain a comprehensive view of ISSRM approaches, we consider four

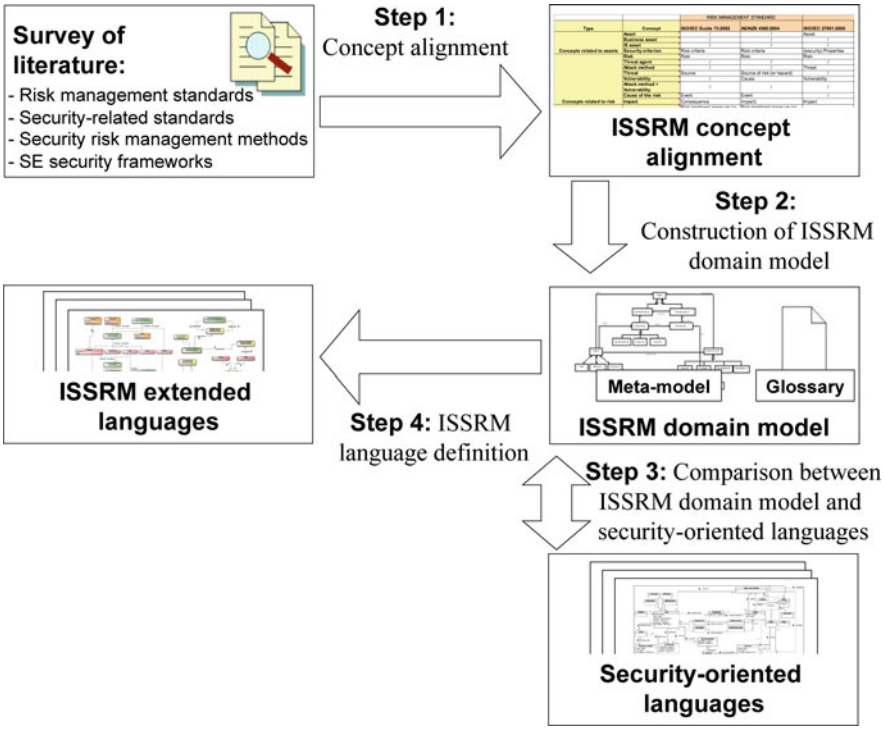

Fig. 1 Research method

Table 1 Alignment of five concepts

References	(1)	(2)	(3)	(4)	(5)
ISO/IEC Guide 73	Risk	Event	Consequence	/	/
AS/NZS 4360	Risk	Event	Consequence Impact	/	/
ISO/IEC 27001	Risk	/	Impact	Threat	Vulnerability
ISO/IEC 13335	Risk	/	Harm	Threat	Vulnerability
Common Criteria	Risk	Threat	Consequence	/	Vulnerability
NIST 800-27 NIST 800-30	Risk	/	Impact	Threat	Vulnerability
EBIOS	Risk	Cause	Impact	/	Vulnerability
MEHARI	Risk Risk scenario	/	Consequence	/	/
OCTAVE	Risk	/	Impact Consequence	Threat	Vulnerability
CRAMM	Risk	/	Loss	Threat	Vulnerability
CORAS	Risk	/	Unwanted incident	Threat scenario	Vulnerability
Haley et al. Moffet and Nuseibeh	Risk	/	Impact	Threat	Vulnerability
Firesmith	Risk	/	Harm	Hazard Threat	Vulnerability

main categories of sources: (*i*) RM standards, (*ii*) security-related standards, (*iii*) security RM methods, and (*iv*) security-oriented RE frameworks.

Step 2 – Construction of the ISSRM domain model. Based on the outcomes of step 1, we define a conceptual model of the ISSRM domain as a UML class diagram, complemented with a glossary obtained by reusing and, when needed, improving the most relevant definitions we found.

Step 3 – Comparison between ISSRM domain model and security-oriented languages. Prominent security-oriented RE languages (KAOS extended to security [29], Abuse Frames [31], Misuse Cases [51], Abuse Case [42] and Secure-Tropos [47]) are confronted with the ISSRM domain model. We investigate the meta-models and definitions of those languages, trying to find out which concepts of the ISSRM domain model are fully supported, partially supported or missing. The main expected results of this step are:

- The validation of the claim that those RE languages overlook RM;
- The assessment of the coverage of each modelling language with respect to ISSRM;
- The identification of the improvements (extensions or revisions) required to make the languages suitable for ISSRM.

Step 4 – Definition of ISSRM language support. As mentioned in the introduction, our final goal is to provide ISSRM-compliant versions of common RE languages. Our aim is to do so by meeting the highest standards in conceptual

language definition [20, 45]. Steps 1–3 are intended to guarantee sound and agreed conceptual foundations. But these are not the only criteria. Hence, step 4 will also address the formal definition of syntax and semantics, which facilitates unambiguous interpretation and automated reasoning. We will also take into account "softer", but equally important properties, such as appropriateness of the graphical symbols and structuring mechanisms.

Further motivations for this research method can be found in [11, 40, 41]. The reader should also note that although this process looks rather sequential, steps 1–4 are meant to be conducted in an iterative and incremental way. In this chapter we focus on the first two steps. In the conclusion, we report on the progress made with steps 3 and 4.

3 Survey of the Literature

The survey of the literature is divided into three parts. The first part (Sect. 3.1) delimits the scope of our survey and provides some basic definitions. The second part (Sect. 3.2) is concerned with ISSRM standards, methods and studies. These sources are used as foundations for the ISSRM domain model (which will be described in Sect. 5). The third part (Sect. 3.3) surveys the security-oriented modelling languages. Those are candidate for comparison and extension according to the ISSRM domain model. However, such comparisons and extensions are out of the scope of the present chapter.

3.1 Scope of the Survey and Basic Definitions

The most generally agreed upon definition of risk is the one found in ISO/IEC Guide 73. There, a risk is defined as a "combination of the probability of an event and its consequence" [22]. Following this definition, RM is defined as "coordinated activities to direct and control an organisation with regard to risk" [22]. Depending on the context, RM can address various kinds of issues [24, 54]. For example, risks can be related to the organisation's management (e.g., illness of a key person in regards to the business), finance (e.g., related to investment), environment (e.g., pollution), or security.

In our research, we focus only on *security* RM. Other kinds of risks, such as financial or project risk, are deemed out of scope. The common denominator of the ISSRM approaches is the fact that there are security objectives to reach (or security properties to respect) to ensure reasonable protection of the organisation's assets. Assets are generally defined as anything that has value to the organisation, and thus needs to be protected. However, we will always look at assets related to an organisation's IS, that is, "[a] system, whether automated or manual, that comprises people, machines, and/or methods organized to collect, process, transmit, and disseminate data that represent user information" [56]. Thus, in a given IS context, assets may

include hardware, software and network as well as people and facilities playing a role in the IS and therefore in its security, e.g., people encoding data, and arguably such things as air conditioning of a server room. All of these are subject to risks and those risks have to be evaluated with respect to the IS properties that could be damaged. Those properties include *confidentiality*, *integrity* and *availability* of information and/or processes in an organisation [23]:

- *Confidentiality* is the property that information is not made available or disclosed to unauthorised individuals, entities, or processes.
- *Integrity* is the property of safeguarding the accuracy and completeness of assets.
- *Availability* is the property of being accessible and usable upon demand by an authorised entity.

Some other criteria like authenticity, non-repudiation or accountability [23] might be added when the context requires, but they are usually deemed secondary. Summing up, the objective of ISSRM is to protect essential constituents of an IS, from all harm to their security (confidentiality, integrity, availability).

3.2 Risk Management Standards, Methods and Studies

The first family of sources that we review are *RM standards*. Those documents typically contain general considerations about RM and form the basis upon which domain-specific RM approaches are built.

- ISO/IEC Guide 73 [22]: This guide defines the RM vocabulary and guidelines for use in ISO standards. It mainly focuses on terminology, which is of great interest with respect to our research method.
- AS/NZS 4360 [3]: This joint Australian/New-Zealand standard provides a generic guide for RM. The document proposes an overview of the RM terminology and process.

The second family of sources consists of (IS and IT) *security standards*. The selected documents often contain a section on security-specific terminology. Sometimes, some RM concepts are mentioned.

- ISO/IEC 27001 [25]: The purpose of this standard is to act as a reference for establishing, implementing, operating, monitoring, reviewing, maintaining and improving an Information Security Management System (ISMS), that is the part of an organisation that is concerned with information security. The principles and terminology related to IS Management System are provided.
- ISO/IEC 13335-1 [23]: This standard is the first of the ISO/IEC 13335 guidelines series that deals with the planning, management and implementation of IT security. It describes concepts and principles of IT security that may be applicable to different organisations.

- Common Criteria [8]: "Common Criteria" (standardised in version 2.3 by ISO/IEC 15408) provides a common set of requirements on the security functions of IT products and systems, and on assurance measures applied to them during a security evaluation. The first part, entitled "Introduction and general model", is the most relevant with respect to our research scope.
- NIST 800-27 Rev A [53]/NIST 800-30 [52]: Among the series of publications proposed by NIST, the 800-series is about computer security. In this series, NIST 800-27 and NIST 800-30 are in our scope. Terminology and concepts are provided by these standards, which are consistent with each other.

Risk management methods are the third family of sources. In 2004, a CLUSIF[1] study inventoried over 200 security RM methods. We select a representative subset of RM methods based on some recent studies, like the report "Inventory of risk assessment and risk management methods" [13] from ENISA. Most of these methods are supported by software tools, but we will concentrate on their methodological part.

- EBIOS [9] The EBIOS method is developed and maintained by the ANSSI in France.
- MEHARI [6] MEHARI is a RM method developed by the CLUSIF and built on the top of two other RM methods: MARION [5] and MELISA [10], not maintained anymore.
- OCTAVE [1]: OCTAVE is an approach to information security risk evaluation developed by the SEI.
- CRAMM [21]: CRAMM is a RM method from the UK, originally developed by CCTA in 1985 and currently maintained by Insight Consulting.
- CORAS [55]: CORAS is the result of a European project that developed a tool-supported framework for risk assessment of security-critical systems.

Finally, the last family consists of *security frameworks* proposed in the scientific literature. Whereas the previous sources were practitioner-oriented, these are more research-oriented. They originate essentially from the RE literature.

- Haley et al. [18, 19] and Moffett and Nuseibeh [44] propose a framework for dealing with security requirements.
- Firesmith [15, 16] presents a set of related information models that provides the theoretical foundation underlying safety and security engineering. A process to effectively deal with both safety and security engineering is also proposed.

A final remark is about SQUARE [43], a stepwise methodology for eliciting, categorising, and prioritising security requirements for IT systems and applications. Although SQUARE is focussed on security RE and suggests using an ISSRM approach to elicit security requirements, it was not retained in this survey because the first step of SQUARE consists in defining the terminology to be used in

[1] http://www.clusif.asso.fr/en/clusif/present/.

the project. Therefore SQUARE does not rely on a pre-defined terminology that we could use.

3.3 State of the Art of Security-Oriented Modelling Languages

Many security modelling languages, or most often security extensions to existing languages, were developed. Existing approaches based on UML have been enriched with security modelling capabilities. In Misuse Cases [51] and Abuse Cases [42], which are extensions of "Use Case" diagrams, the focus is on elicitation of new threats and vulnerabilities that could be exploited by malicious actors. SecureUML [35] extends several UML diagrams. The approach focuses on authorisation constraints and its goal is to automatically generate complete access control infrastructures. UMLsec [28] is a UML profile that allows adding security-related information to UML diagrams. Both SecureUML and UMLsec address security at the design level. They, thus, do not focus on business assets and high-level security requirements.

The KAOS goal-oriented framework addresses security concerns by treating attacks as anti-goals [29]. Anti-goals are the attacker's goals and generate obstacles to security goals. Extensions of the *i** goal-oriented framework [57] also address security problems. For instance, Liu et al. [34] represent attacks as tasks with negative contributions to security softgoals. A formalisation of *i** to deal with security issues is proposed in Secure-Tropos [17, 47]. It suggests, first, to extend the concepts and the processes of *i**/Tropos and, then, to integrate techniques such as security reference diagrams and security attack scenarios. Recently, additional work [12] has been done on representing the notion of vulnerability in *i**. Asnar et al. introduced the Tropos Goal-Risk Framework [2] that addresses RM at three different levels, combining together asset, risk, and risk treatment views. However, the Tropos Goal-Risk framework does not focus on IS security, but supports the concept of risk in general, including project management risk and financial risk, for instance. Finally, Problem Frames extensions were also proposed to handle security issues. Anti-requirements were introduced by Abuse Frames [33]. Abuse Frames are used to delimit the scope of a security problem and thereby are meant to facilitate the analysis of threats and vulnerabilities as well as the elicitation of security requirements. In future work, we plan to confront the concepts of these languages with the concepts of the ISSRM domain.

4 ISSRM Concept Alignment

4.1 Concepts to Consider

The first task of the concept alignment phase is to define the range of concepts to study. In [14], a comparison between the concepts used in various security RE

methods was proposed. Our work has a different scope, that is, ISSRM. Here, the core concept to consider is *risk*. Yet, risk is not an isolated concept. A risk (i) depends on the *security needs* placed on the IS *assets* and (ii) is the subject of *risk treatments*. These are the concepts that we include in our first iteration on step 1, but our scope is likely to expand along the way. Conversely, specific usages of our concept alignment table could consider only subsets of it if not all concepts are needed.

4.2 Overview of the Alignment Table

In this section, we analyse the concept of *risk* starting from the definitions found in the sources listed in Sects. 3.1 and 3.2. We focussed on RM standards and security standards; RM methods and RE security frameworks are addressed in [38]. Content-wise, we focus on the notion of risk and its associated components. Risk-related metrics [9, 15, 52] like, for example, its value or its likelihood, are currently not considered.

4.2.1 Risk Management Standards

ISO Guide 73 gives the following definition of a risk:

> **Risk:** combination of the probability of an event and its consequence.

AS/NZS 4360 proposes a similar definition in its glossary:

> **Risk:** the chance of something happening that will have an impact on objectives
> NOTE 1: A risk is often specified in terms of an event or circumstance and the consequences that may flow from it.

Both sources indicate that a risk is composed of two related elements: a cause, called event or "something happening"; and a consequence, also called impact. This consideration is valid in all risk-related domains. To refine our analysis, we compare the above definitions with the ones from the security domain.

4.2.2 Security Related Standards

In ISO/IEC 27001 [25], the concept of risk is not present in the glossary, but in an excerpt of the standard presenting the risk identification step, we find:

> Identify the **risks**.
>
> 1) Identify the assets within the scope of the ISMS, and the owners of these assets.
> 2) Identify the threats to those assets.
> 3) Identify the vulnerabilities that might be exploited by the threats.
> 4) Identify the impacts that losses of confidentiality, integrity and availability may have on the assets.

In ISO/IEC 13335 [23], a risk is defined in the glossary in terms of three related concepts:

> **Risk:** the potential that a given threat will exploit vulnerabilities of an asset or group of assets and thereby cause harm to the organization.

The analysis of both sources [23, 25], and mainly the definition from [23] which is more explicit than the succession of steps presented in [25], shows that these definitions of a risk are compliant with RM standards, because a risk is always composed of a *cause* and a *consequence*. However, the definitions introduce some new concepts: the cause of the risk is presented as the combination of *threat* and *vulnerability*, and the *consequence* is considered as the *impact* or *harm* (see Table 1). The concept of *asset*, which is not analysed in depth in this section, is also introduced as related to the notion of risk. It is defined as anything that has value to the organisation [23]. Common Criteria (CC) [8] defines risk with a finer granularity:

> Threats are categorised as the potential for abuse of protected assets. The CC characterises a threat in terms of a threat agent, a presumed attack method, any vulnerabilities that are the foundation for the attack, and identification of the asset under attack. An assessment of **risks** to security would qualify each threat with an assessment of the likelihood of such a threat developing into an actual attack, the likelihood of such an attack proving successful, and the consequences of any damage that may result. A threat shall be described in terms of an identified threat agent, the attack, and the asset that is the subject of the attack. Threat agents should be described by addressing aspects such as expertise, available resources, and motivation. Attacks should be described by addressing aspects such as attack methods, any vulnerabilities exploited, and opportunity.

Here the cause of the risk is called threat and it encompasses vulnerability, unlike [25] and [23] that define them as related, but separate concepts at the same level. The threat in [8] has multiple sub-components like *threat agent, attack method, attack*, etc. Details of those sub-components can be found in [40]. Threat in ISO/IEC 27001 or ISO/IEC 13335 has thus not the same sense as threat in CC, which is equivalent to the global cause of the risk, encompassing threat and vulnerability. Threat from [23, 25] and threat from [8] are thus not aligned in Table 1. NIST standards also propose a different definition for a risk [52, 53]:

> **Risk:** The net mission/business impact considering (1) the likelihood that a particular threat source will exploit, or trigger, a particular information system vulnerability and (2) the resulting impact if this should occur.

Here, risk is once again defined with the help of three components: *threat source, vulnerability* and *impact*. The concept of threat is defined as the combination of a threat source, its motivation (for human threat) and threat actions, like hacking, social engineering, or system intrusion [52].

The use of the term risk in security related standards is more precise than in RM standards, but remains compliant with the latter. It is thus a mere specialisation of the term. The concept of risk is therefore aligned between the sources in Table 1. However the precision of the components of a risk increases. The consequence of the risk differs only in how it is named (*consequence*, *impact* or *harm*)

but the semantics remains largely the same. However, the cause of the risk is presented as a composition of elements, which are different depending on the sources. Differences and equivalences are shown in Table 1.

The concept of *asset* is often mentioned in the definition of risk found in security related standards. It is sometimes associated with *threat* [25], sometimes with *vulnerability* [23] and sometimes with *attack* [8]. In any case, the concept of *asset* plays a role in the definition of risk and should be linked to it. However, due to page limits, we cannot go into such details here. More details can be found in [38].

5 ISSRM Domain Model

The first step of the method has resulted in an alignment of the ISSRM concepts, found in the literature. The second step of the method includes the construction of the ISSRM domain model, presented in Fig. 2. For each concept of the alignment table, a name is chosen. Then, concepts are linked based on the relationships identified in [39]. A glossary is provided together with the domain model, giving a definition for each of its concepts. In this section we introduce the main concepts and their definitions. They are illustrated by examples related to an architecture engineering company [38]. The ISSRM domain model features three principal groups of concepts: (i) *asset*-related concepts, (ii) *risk*-related concepts, and (iii) *risk treatment*-related concepts.

Asset-related concepts describe what are the important assets to protect, and what are the criteria to guarantee asset security. The concepts are:

Asset – anything that has value to the organisation and is necessary for achieving its objectives. Examples: *technical plan; structure calculation process; architectural competence; operating system; Ethernet network; people encoding data; system administrator; air conditioning of server room.*

Note: This concept is the generalisation of the business asset and IS asset concepts.

Business asset – information, process, skill inherent to the business of the organisation that has value to the organisation in terms of its business model and is necessary for achieving its objectives. Examples: *technical plan; structure calculation process; architectural competence.*

Note: Business assets are immaterial.

IS asset – a component or part of the IS that has value to the organisation and is necessary for achieving its objectives and supporting business assets. An IS asset can be a component of the IT system, like hardware, software or network, but also people or facilities playing a role in the IS and therefore in its security. Examples: *operating system; Ethernet network; people encoding data; system administrator; air conditioning of server room.*

Note 1: IS assets are (with the exception of software) material.

Note 2: Sometimes, for conducting a macroscopic analysis, it is necessary to define a system composed of various IS assets as an IS asset.

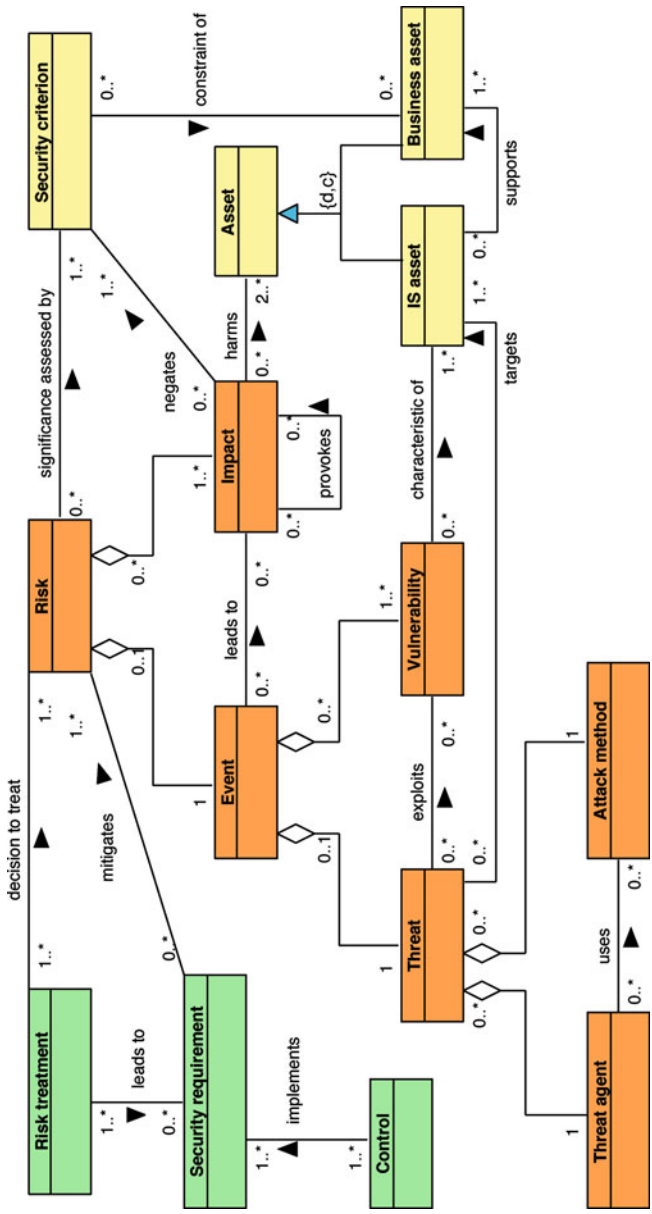

Fig. 2 ISSRM domain model

Security criterion (also called *security property*) – property or constraint on business assets that characterises their security needs. Security criteria act as indicators to assess the significance of a risk. Examples: *confidentiality; integrity; availability; non-repudiation; accountability.*

Note: The security objectives of an IS are defined using security criteria on business assets (e.g., confidentiality of the technical plans; integrity of the structure calculation process).

Our second group of concepts are *risk-related concepts*. They present how the risk itself and its immediate components are defined.

Risk – the combination of a threat with one or more vulnerabilities leading to a negative impact harming one or more of the assets. Threat and vulnerabilities are part of the risk event and impact is the consequence of the risk. Examples: *a hacker using social engineering on a member of the company, because of weak awareness of the staff, leading to unauthorised access to personal computers and loss of integrity of the structure calculation process; a thief entering a company building thanks to deficient physical access control, stealing documents containing sensitive information and thereby provoking loss of confidentiality of technical plans*.

Impact – the potential negative consequence of a risk that may harm assets of a system or an organisation, when a threat (or an event) is accomplished. The impact can be described at the level of IS assets (data destruction, failure of a component, etc.) or at the level of business assets, where it negates security criteria, like, for example, loss of confidentiality of an information, loss of integrity of a process, etc. Examples: *password discovery (IS level); loss of confidentiality of technical plans (business level*).

Note: An impact can provoke a chain reaction of impacts (or indirect impacts), like for example a loss of confidentiality on sensitive information leads to a loss of customer confidence.

Event – the combination of a threat and one or more vulnerabilities. Examples: *a hacker using social engineering on a member of the company, exploiting weak awareness of the staff; a thief entering a company building thanks to deficient physical access control.*

Note: Event is a generic term, used pervasively in RM and defined as the "occurrence of a particular set of circumstances" [22]. The definition provided in this glossary is specific to IS security.

Vulnerability – the characteristic of an IS asset or group of IS assets that can constitute a weakness or a flaw in terms of IS security. Examples: *weak awareness of the staff; deficient physical access control; lack of fire detection*.

Threat – potential attack, carried out by an agent that targets one or more IS assets and that may lead to harm to assets. A threat is constituted of a threat agent and an attack method. Examples: *a hacker using social engineering on a member of the company; a thief entering a company building and stealing media or documents.*

Threat agent – an agent that can potentially cause harm to assets of the IS. A threat agent triggers a threat and is thus the source of a risk. Examples: *staff members with little technical skills and time but possibly a strong motivation to*

carry out an attack; hacker with considerable technical skills, well equipped and strongly motivated by the money he could make.

Note: A threat agent can be characterised by expertise, available resources and motivation.

Attack method – standard means by which a threat agent carries out a threat. Examples: *system intrusion; theft of media or documents*.

Risk treatment-related concepts describe what decisions, requirements and controls should be defined and implemented in order to mitigate possible risks. The different risk treatment-related concepts are different levels of design decisions on the IS.

Risk treatment – the decision of how to treat the identified risks. A treatment satisfies a security need, expressed in generic and functional terms, and can lead to security requirements. Categories of risk treatment decisions include:

- *Avoiding* the risk (risk avoidance decision) – decision not to become involved in, or to withdraw from, a risk. Functionalities of the IS are modified or discarded for avoiding the risk;
- *Reducing* the risk (risk reduction decision) – action to lessen the probability, negative consequences, or both, associated with a risk. Security requirements are selected for reducing the risk;
- *Transferring* the risk (risk transfer decision) – sharing with another party the burden of loss from a risk. A third party is thus related to the (or part of the) IS, ensuing sometimes some additional security requirements about third parties;
- *Retaining* the risk (risk retention decision) – accepting the burden of loss from a risk. No design decision is necessary in this case.

Examples: *not connecting the IS to the Internet (risk avoidance); taking measures to avoid network intrusions (risk reduction); taking an insurance for covering a loss of service (risk transfer); accepting that the service could be unavailable for 1 hour (risk retention).*

Note: Risk treatment is basically a shortcut for risk treatment decision, according to the state of the art.

Security requirement – a condition over the phenomena of the environment that we wish to make true by installing the IS, in order to mitigate risks. This definition is inspired from [26]. Examples: *appropriate authentication methods shall be used to control access by remote users; system documentation shall be protected against unauthorised access.*

Note 1: Risk reduction decisions lead to security requirements. Sometimes, risk transfer decisions need some security requirements about third parties. Avoiding risk and retaining risk do not need any security requirement.

Note 2: Each security requirement contributes to cover one or more risk treatments for the target IS.

Control (also called *countermeasure* or *safeguard*) – a designed means to improve security, specified by a security requirement, and implemented to comply with it. Security controls can be processes, policies, devices, practices or other

actions or components of the IS and its organisation that act to reduce risks. Examples: *firewall; backup procedure; building guard*.

6 Conclusion

Today, support for security risk management cannot be overlooked anymore, especially during the early phases of IS development. A review of the state of the art indicates that practitioner-oriented standards under-exploit modelling techniques. On the other hand, RE modelling techniques tend to neglect RM, and thereby the cost-effectiveness concerns that are important to practitioners. To improve on this situation, we aim at extending RE languages with ISSRM concepts. In this chapter, we reported on an important step towards this goal: the elaboration of a domain model for ISSRM. This approach is in line with the practices advocated since long time by pioneers of the IS modelling discipline [50].

The proposed domain model extends an earlier version [40]. It consists of a conceptual model (UML class diagram) that highlights the main ISSRM concepts and their relationships, together with their corresponding definitions. Preliminary validation [19] of this domain model has already been performed by practitioners, researchers and standardization experts. We also obtained feedback on usage of the domain model as a teaching artefact for an ISO/IEC 27001 certification. Additionally, encouraging results were also obtained with students involved in a professional Information System Security Management Master programme.

Our on-going work includes enriching the domain model with various metrics commonly used for risk estimation and evaluation [38]. Finally, our current work is progressing according to the steps 3–4 of the research method presented in Sect. 2. With respect to step 3, we started evaluating existing security-oriented RE languages with the intent to later extend them for better supporting ISSRM. At this time, we have analysed KAOS [38], Misuse cases [36] and Secure Tropos [37]. Regarding step 4, en extension of Secure Tropos is under way.

Acknowledgments Thanks to Germain Saval for his help in editing this chapter. And finally, we would like to express our immense gratitude to Colette Rolland for showing us the way.

References

1. Alberts CJ, Dorofee AJ (2001) OCTAVE method implementation guide version 2.0. Carnegie Mellon University, Software Engineering Institute, Pittsburgh, PA
2. Asnar Y, Giorgini P (2006) Modelling risk and identifying countermeasure in organizations. In: Proceedings of the 1st interational workshop on critical information intrastructures security (CRITIS'06), Springer, Berlin, pp 55–66
3. AS/NZS 4360 (2004) Risk management. SAI Global
4. Bresciani P, Giorgini P, Giunchiglia F, Mylopoulos J, Perin, A (2004) TROPOS: an agent-oriented software development methodology. Autonomous Agents Multi-Agent Systems 8:203–236

5. CLUSIF (1998) MARION (Méthodologie d'Analyse des Risques Informatique et d'Optimation par Niveau) available at http://www.clusif.asso.fr
6. CLUSIF (2007) MEHARI 2007: concepts and mechanisms. http://www.clusif.asso.fr/fr/production/ouvrages/pdf/CLUSIF-risk-management.pdf. Last Accessed 21 Feb 2010
7. Cockburn A (2001) Writing effective use cases. Addison-Wesley Longman Publishing Co., Boston, MA, USA
8. Common Criteria version 2.3 (2005) Common criteria for information technology security evaluation, CCMB-2005-08-002. http://www.tse.org.tr/turkish/belgelendirme/ortakkriter/ccpart2v2.3.pdf. Last Accessed 21 Feb 2010
9. DCSSI (2004) EBIOS – expression of needs and identification of security objectives. http://www.ssi.gouv.fr/archive/en/confidence/ebiospresentation.html. Last Accessed 21 Feb 2010
10. Direction des Constructions Navales (1989) MELISA (Méthode d'Evaluation de la Vulnérabilité Résiduelle des Systèmes d'Information). Paris, France
11. Dubois E, Mayer N, Rifaut A, Rosener V (2006) Contributions méthologiques pour l'amélioration de l'analyse des risques. In: Enjeux de la sécurité multimédia (Traité IC2, série Informatique et systèmes d'information). Hermes Science Publications, Paris, pp 79–131
12. Elahi G, Yu E, Zannone N (2010) A vulnerability-centric requirements engineering framework: analyzing security attacks, countermeasures, and requirements based on vulnerabilities. Reqs Eng Journal 15(1):41–62
13. ENISA (European Network and Information Security Agency) (2006) Inventory of risk assessment and risk management methods. http://www.enisa.europa.eu/act/rm/files/deliverables/inventory-of-risk-assessment-and-risk-management-methods. Last Accessed 21 Feb 2010
14. Fabian B, Gürses S, Heisel M, Santen T, Schmidt H (2010) A comparison of security requirements engineering methods. Reqs Eng Journal 15(1):7–40
15. Firesmith DG (2003) Common concepts underlying safety, security, and survivability engineering. CMU/SEI-2003-TN-033 Carnegie Mellon University, Software Engineering Institute, Pittsburgh, PA
16. Firesmith DG (2007) Engineering safety and security related requirements for software intensive systems. In: Companion to the proceedings of the 29th international conference on software engineering (COMPANION'07). IEEE Computer Society, p 169
17. Giorgini P, Massacci F, Zannone N (2005) Security and trust requirements engineering. In: Foundations of security analysis and design III. LNCS, vol 3655. Springer, pp 237–272
18. Haley CB, Laney RC, Moffett JD, Nuseibeh B (2008) Security requirements engineering: a framework for representation and analysis. IEEE Trans Softw Eng 34:133–153
19. Haley CB, Moffett JD, Laney RC, Nuseibeh B (2006) A framework for security requirements engineering. In: Proceedings of the 2nd international workshop on software engineering for secure systems (SESS'06), ACM, pp 35–42
20. Harel D, Rumpe B (2004) Meaningful modeling: what's the semantics of "semantics"? Computer 37:64–72
21. Insight Consulting (2003) CRAMM (CCTA Risk Analysis and Management Method) User Guide version 5.0. SIEMENS
22. ISO/IEC Guide 73 (2002) Risk management – vocabulary – guidelines for use in standards. International Organization for Standardization, Geneva
23. ISO/IEC 13335-1 (2004) Information technology – security techniques – management of information and communications technology security – part 1: concepts and models for information and communications technology security management. International Organization for Standardization, Geneva
24. ISO 14001 (2004) Environmental management systems – requirements with guidance for use. International Organization for Standardization, Geneva
25. ISO/IEC 27001 (2005) Information technology – security techniques – information security management systems – requirements. International Organization for Standardization, Geneva

26. Jackson M (1995) Software requirements & specifications: a lexicon of practice, principles and prejudices. ACM/Addison-Wesley, New York
27. Jackson M (2001) Problem frames: analyzing and structuring software development problems. Addison-Wesley, New York
28. Jürjens J (2002) UMLsec: extending uml for secure systems development. In: Proceedings of the 5th international conference on the unified modeling language (UML'02). LNCS, vol 2460. Springer, pp 412–425
29. van Lamsweerde A (2004) Elaborating security requirements by construction of intentional anti-models. In: Proceedings of the 26th international conference on software engineering (ICSE'04), IEEE Computer Society, pp 148–157
30. van Lamsweerde A, Letier E (2000) Handling obstacles in goal-oriented requirements engineering. IEEE Trans Softw Eng 26:978–1005
31. Lin L, Nuseibeh B, Ince D, Jackson M (2004) Using abuse frames to bound the scope of security problems. In: Proceedings of the 12th IEEE international conference on requirements engineering (RE'04), IEEE Computer Society, pp 354–355
32. Lin L, Nuseibeh B, Ince D, Jackson M, Moffett JD (2003) Analysing security threats and vulnerabilities using abuse frames. Technical report No: 2003/10, Open University
33. Lin L, Nuseibeh B, Ince D, Jackson M, Moffett JD (2003) Introducing abuse frames for analysing security requirements. In: Proceedings of the 11th IEEE international conference on requirements engineering (RE'03), IEEE Computer Society, pp 371–372
34. Liu L, Yu E, Mylopoulos J (2003) Security and privacy requirements analysis within a social setting. In: Proceedings of the 11th IEEE international conference on requirements engineering (RE'03), IEEE Computer Society, p 151
35. Lodderstedt T, Basin D, Doser J (2002) SecureUML: a UML-based modeling language for model-driven security. In: Proceedings of the 5th international conference on the unified modeling language (UML'02), Springer, pp 426–441
36. Matulevičius R, Mayer N, Heymans P (2008) Alignment of misuse cases with security risk management. In: Proceedings of the 3rd international conference on availability, reliability and security (ARES'08), IEEE Computer Society, pp 1397–1404
37. Matulevičius R, Mayer N, Mouratidis H, Dubois E, Heymans P, Genon N (2008) Adapting secure tropos for security risk management during early phases of the information systems development. In: Proceedings of the 20th international conference on advanced information systems engineering (CAiSE'08). LNCS, vol 5074. Springer, pp 541–555
38. Mayer N (2009) Model-based management of information system security risk. PhD thesis, University of Namur
39. Mayer N, Genon N (2006) Design of a modelling language for information system security risk management –elicitation of relationships between concepts and meta-model of each source. Technical report. University of Namur
40. Mayer N, Heymans P, Matulevičius R (2007) Design of a modelling language for information system security risk management. In: Proceedings of the 1st international conference on research challenges in information science (RCIS'07), IEEE Xplore Digital Library, pp 121–132
41. Mayer N, Rifaut, A, Dubois E (2005) Towards a risk-based security requirements engineering framework. In: Proceedings of the 11th international workshop on requirements engineering: foundation for software quality (REFSQ'05), Springer, pp 83–97
42. McDermott J, Fox C (1999) Using abuse case models for security requirements analysis. In: Proceedings of the 15th annual computer security applications conference (ACSAC'99), IEEE Computer Society, pp 55–65
43. Mead NR, Hough ED, Stehney TR (2005) Security quality requirements engineering (SQUARE) methodology. Technical report CMU/SEI-2005-TR-009, ESC-TR-2005-009Carnegie Mellon University – Software Engineering Institute, Pittsburgh, PA
44. Moffett JD, Nuseibeh B (2003) A framework for security requirements engineering. Report YCS 368 Department of Computer Science, University of York, UK

45. Moody DL (2009) Evidence-based notation design: towards a scientific basis for constructing visual notations in software engineering. IEEE Trans Softw Eng 35(6):756–779
46. Mouratidis H, Giorgini P (2010) Extending i* and tropos to model security. In: Yu E, Giorgini P, Maiden N, Mylopoulos J (eds) Social modeling for requirements engineering. MIT (in press), Cambridge, *Massachusetts* (USA)
47. Mouratidis H, Giorgini P, Manson GA, Philp I (2002) A natural extension of tropos methodology for modelling security. In: Proceedings of the agent oriented methodologies workshop (OOPSLA'02)
48. Oladimeji EA, Supakkul S, Chung L (2006) Security threat modeling and analysis: a goal-oriented approach. In: Proceedings of the 10th international conference on software engineering and applications (SEA'06), pp 178–185
49. Olle TW, Hagelstein J, Macdonald IG., Rolland C, Sol HG, Van Assche FJM, Verrijn-Stuart AA (1992) Information systems methodology: a framework for understanding, 2nd edn. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA
50. Rolland C (1998) An information system methodology supported by an expert design tool. Elsevier Science, University of Paris
51. Sindre G, Opdahl AL (2004) Eliciting security requirements with misuse cases. Reqs Eng J 10(1):34–44
52. Stoneburner G, Goguen A, Feringa A (2002) NIST special publication 800-30: risk management guide for information technology systems. National Institute of Standards and Technology, Gaithersburg
53. Stoneburner G, Hayden C, Feringa A (2004) NIST special publication 800-27 rev. A: engineering principles for information technology security (a baseline for achieving security). National Institute of Standards and Technology, Gaithersburg
54. The Project Management Institute (2001) Project management body of knowledge www.pmi.org/
55. Vraalsen F, Mahler T, Lund MS, Hogganvik I, den Braber F, Stølen K (2007) Assessing enterprise risk level: the CORAS approach. In: Khadraoui D, Herrmann F (eds) Advances in enterprise information technology security. Idea Group, IGI Global, Hershey, Pennsylvania pp 311–333
56. Wikipedia (2008) Information system definition. http://en.wikipedia.org/wiki/Information_system
57. Yu E (1996) Modelling strategic relationships for process reengineering. PhD Thesis, University of Toronto, Toronto, ON, Canada