Raimundas Matulevičius

Fundamentals of Secure System Modelling

Springer

Chapter 2
Domain Model for Information Systems Security Risk Management

One important task during secure systems development is to understand what assets need to be protected against which risks, and how these risks could be mitigated by proposed security countermeasures. However, the problem is that the domain terminology needs to be understood in the same way by the people working on the problem. To solve this problem, some domain model should be introduced to guide these activities.

A domain model for information systems security risk management (ISSRM) is developed through a survey of security-related standards, security risk management standards, and security risk management methods. This domain model is presented in [53, 133]. In this chapter we present the concepts of the ISSRM domain model, risk management process and security risk metrics. We will complete the chapter with the illustrative example and some overview of the related approaches for the security risk management.

2.1 Domain Model

The ISSRM domain model [53, 133] consists of three major groups of concepts: asset-related concepts (see Sect. 2.1.1), risk-related concepts (see Sect. 2.1.2), and risk treatment-related concepts (see Sect. 2.1.3). In Fig. 2.1, the domain model is presented as a UML class diagram and a glossary where concept definitions are provided. In this section we will briefly present these definitions.

2.1.1 Asset-Related Concepts

Asset-related concepts describe which of an organisation's assets are important to protect and what criteria guarantee a certain level of asset security. An *asset* is anything that is valuable and plays a role accomplishing the organisation's

R. Matulevičius, *Fundamentals of Secure System Modelling*,
DOI 10.1007/978-3-319-61717-6_2

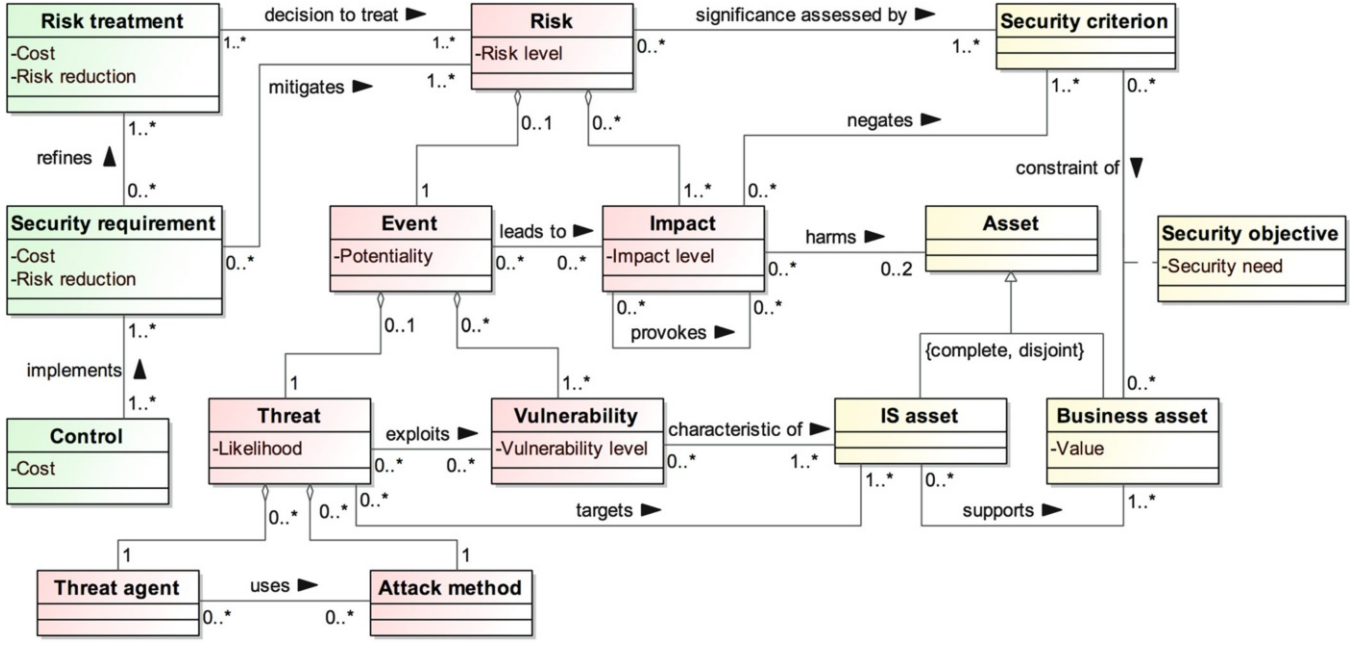

Fig. 2.1 The ISSRM domain model, adapted from [53, 133]

objectives. Assets can be classified as business assets or organisational assets. A *business asset* describes the information, processes, capabilities and skills essential to the business and its core mission. Typically, business assets are immaterial. An *IS asset* is a component or part of an information system, valuable to the organisation since it supports business assets. An IS asset[1] can be a component of the information technology system (e.g., hardware, software or network), but also person or a facility that plays a role in the system and, therefore, in its security. The IS assets (with the exception of software) are material. A *security criterion* (also called *security property*) characterises a security need and is a property or constraint on business assets. The security objectives are defined using security criteria on business assets. Thus, the security criteria describe the security needs, which are, typically, expressed as confidentiality, integrity and availability of business assets.

2.1.2 Risk-Related Concepts

Risk-related concepts introduce definitions of risk itself and its immediate components. A *risk* is a combination of a threat with one or more vulnerabilities leading to a negative impact on two or more assets[2] by harming them. The combination of threat and vulnerabilities represents a risk event and impact is the consequence of this risk.

An *impact* is the potential negative consequence of a risk that negates the security criterion defined for business assets and harms these assets when a threat (or an event) is accomplished. The impact can also be described at the level of IS assets (e.g., data destruction, failure of a component, etc.) or at the level of business assets, where it negates security criteria (e.g., loss of information confidentiality, loss of process integrity, loss of data availability). In addition, one impact can provoke a chain reaction of impacts (or indirect impacts), for example, a loss of confidentiality of sensitive information leads to a loss of customer confidence.

A risk *event* is an aggregation of a threat and one or more vulnerabilities. A *vulnerability* is the characteristic of an IS asset or group of IS assets that exposes a weakness or flaw in terms of security. A *threat* is an incident initiated by a threat agent using an attack method to target one or more IS assets by exploiting their vulnerabilities. A *threat agent* is an agent that has the means to intentionally harm IS assets. A threat agent triggers a threat and, thus, is the source of a risk. The threat agent is characterised by expertise, its available resources, and motivation. An *attack method* describes a standard means by which a threat agent executes a threat.

[1] Later in the book we will refer to the *IS asset* as a *system asset*, indicating the generosity of different IS components in supporting the business assets.

[2] The domain model assumes that if there is a business asset, it is supported by at least one IS asset. Thus, the impact harms at least one business asset and at least one IS asset.

2.1.3 Risk Treatment-Related Concepts

Risk-treatment related concepts describe the concepts to treat risk. A *risk treatment decision* is a decision to treat the identified risk. A treatment satisfies a security need, expressed in generic and functional terms and refined to security requirements. There are four categories of risk treatment decisions possible:

- *Risk avoidance* is a decision not to become involved with or to withdraw from a risk. The system's functionality is modified or discarded for avoiding the risk;
- *Risk reduction* includes actions to lessen the probability, negative consequence, or both associated with a risk. Security requirements are, typically, selected for reducing the risks;
- *Risk transfer* defines how risk parties could share the burden of loss from a risk. A third party is related to the (or part of the) system. This also means that some security requirements could be defined regarding the third party;
- *Risk retention* constitutes acceptance of the burden of loss from a risk. No design decision is necessary in this case.

A *security requirement* is a condition on the phenomena of the environment that we wish to make true by installing the information system, in order to mitigate risks. A security requirement is the refinement of a risk treatment decision to mitigate the risks. On the one hand risk reduction decisions lead to security requirements. But sometimes, risk transfer decisions need to improve some security requirements on third parties. Avoiding risk and retaining risk do not need any security requirement. On the other hand, each security requirement contributes to cover one or more risk treatments for the target system.

A *control* (*countermeasure* or *safeguard*) is a designed means to improve the security by implementing the security requirements. Security controls can be processes, policies, devices, practices or other actions or components of the IS and its organisation that act to reduce risks.

2.2 Relationships and Multiplicities

In [133], relationships and multiplicities between the ISSRM concepts are discussed. They are presented in Fig. 2.1 and discussed in this section.

Relationships of Asset-Related Concepts Assets can be *specialised* as two different kinds: business assets and IS assets. The specialisation is *disjoint* and complete. An information system asset can *support* one or more business assets, but a business asset can have no support in the system (e.g., the selling skills of the sales department are an asset of the company, but they are not part of the system). However, a usual situation is that a business asset *is supported by several* IS assets. Each business asset can *be constrained* from zero (e.g., if the business asset has no

support in the IS) to several security criteria. A security criterion can *be a constraint of* several different business assets, or not constrain any of them.

One or several security criteria can be taken into account to *assess the significance of* a risk. But a security criterion can *be concerned* by none of the risks in the case where there is no relevant impact for this criterion found.

Relationships of Risk-Related Concepts A risk is *composed of* an event and one or more impacts. The same impact can be part of several risks, but an event identifies a given risk. A given event *leads to* zero (if no relevant impact is found; in this case the event does not produce a risk) or to several impacts. An impact can *be caused by* many different events. One or several impacts *can provoke* some other (indirect) impacts.

Impacts *harm* assets, both at the business and at the system levels. An asset can *be harmed* by zero (if no impact is considered as relevant) or several impacts, and an impact *harms* at least one system asset and at least one business asset. At the level of business assets, an impact *negates* one or more security criteria, and a given security criterion can *be negated by* zero (if no relevant impact is concerned with this security criterion) or several impacts.

The risk event *is composed* of a threat and one or more vulnerabilities. A given threat can only be related to a given event. The threat *exploits* zero to several vulnerabilities. If a threat is identified, but has no relevant associated vulnerability, it will be neither part of an event nor a risk. A given vulnerability can *be exploited by* many different threats and, therefore, related to many different events, or *not be exploited by* any of them, if no relevant threat is found.

A vulnerability is a *characteristic of* a system asset or group of them. An IS asset can have from zero to several vulnerabilities. A threat *targets* one or more system assets and this asset can be targeted by zero to several threats.

A threat is *defined in terms* of a threat agent that *uses* an attack method. Each threat agent (respectively, attack method) identified as relevant can be involved in several threats, or sometimes in none of them if no relevant corresponding attack method (respectively threat agent) is found. A given threat agent *uses* from zero to several attack methods, and an attack method can *be used by* zero or more threat agents.

Relationships of Risk Treatment-Related Concepts A risk treatment *expresses* the decision to treat one or more risks. Each identified risk *has a risk treatment* and sometimes several of them can be combined (they are not mutually exclusive). A risk treatment decision *is refined to* one or more security requirements. However, the risk treatments of acceptance and avoidance *are refined to* none security requirement. Each security requirement *refines* one or many risk treatments.

A security requirement *mitigates* one or more risks. A given risk *cannot be mitigated* by any security requirement (e.g., when the risk is accepted), but can be *mitigated by* several security requirements if they are necessary to reach an acceptable level of risk. Finally, a control *implements* one or more security requirements, and the same security requirement may *be implemented by* one or several controls.

2.3 Metrics

The ISSRM security metrics are defined in [135] and included in the visual domain model representation, in Fig. 2.1. The value of business assets is measured using the *Value* metric. This value metric is used to estimate the security need of each business asset in terms of confidentiality, integrity and availability. A metric to assess *Security need* expresses the importance of the security criterion with respect to the business asset. This metric is introduced as an attribute of the security objective concept.

Risk is estimated using the *Risk level* metric. The risk level depends on the event *Potentiality* and the *Impact level*, these two concepts composing the one of risk. Since an event is composed of threat and vulnerability, an event's *Potentiality* is estimated through threat *Likelihood* and *Vulnerability level*. It is necessary to note that threat agent and attack method do not have their own metric representing their level. Some characteristics of threat agents and attack methods can be identified independently, like the motivation and the competence of the threat agent and the kind of attack method (natural, human, etc.). But they can be used as indicators to well estimate the risk-related concepts and mainly the likelihood of a threat.

In risk treatment-related concepts, risk treatment and security requirements are estimated in terms of *Risk reduction* performed and *Cost* incurred. Controls can be only estimated in terms of *Cost*.

The proposed metrics are rather abstract. Their implementation could result in qualitative, quantitative, or combined (i.e., qualitative and quantitative) approaches. *Quantitative risk analysis* typically employs a set of methods or principles for assessing risks, based on the use of precise numbers [135]. It tries to assign hard financial values to assets, expected losses, and cost of controls. This method gives the most accurate data, but quantitative risk analysis requires significant amount of information and time.

Qualitative risk analysis typically employs a set of methods or principles based on non-numerical categories or levels (e.g., very low, low, moderate, high, very high) [135]. The basic process for risk assessment of a qualitative approach is similar to what happens in the quantitative approach. However, the main difference is in the relative calculation values; these do not require a lot of time or staff to calculate precise financial numbers for asset valuation, possible impact from a risk being realised or the cost of implementing controls. The drawback of a qualitative approach is that the results are vague and imprecise because of the relative values.

2.4 Process

The ISSRM domain model is focussed around a process, describing activities to perform in order to manage security risks. In [133] this process (see Fig. 2.2) is reported as the result of the analysis of the security and security risk management standards (see Chap. 1).

The process begins with (a) a study of the organisation's context and the identification of its assets. In this step, the organisation, its environment and the system(s) used in this organisation are described. Then, based on the level of protection required for the assets, one needs to determine the (b) security objectives. Security objectives are defined in terms of confidentiality, integrity and availability of the business assets. The next step of the process is (c) risk analysis, where security risks which harm assets and threaten security objectives are elicited and assessed. Once risk assessment is finished, decisions about (d) risk treatment are taken (e.g., avoiding, reducing, transferring or accepting the risk). The next step (e)) is the elicitation of security requirements to mitigate the identified risks. Finally, security requirements are implemented as security controls (f), i.e., system-specific countermeasures that are implemented within the organisation.

The ISSRM process is iterative. Several iterations need to be performed, until we reach an acceptable level for each risk. Figure 2.2 presents only few the major iteration points. Even after reaching an acceptable level for all risks, the ISSRM process should be regularly reviewed. Risks are obviously not static and should be monitored either automatically or manually by the risk analyst in an organisation. Each modification in the organisation's business, in its context and/or in its system can produce modifications on risks and their levels. In an ideal way, the ISSRM process should in fact be continuously performed, in order to keep the organisation's business and its associated security needs aligned with the taken measures, thus ensuing the required security level.

2.5 ISSRM Application Example

The example of the application of the ISSRM domain model is taken from the *Football Federation case*, described in Sect. 1.7. We will illustrate how the example could be developed using the ISSRM concepts (see Table 2.1 for the asset-oriented concepts).

Let's assume that the Umpire submits the Game report data to the ERIS system. Thus, here the Game report corresponds to the *business asset*, which is supported at least by two *system assets—input interface* (e.g., frame to Submit game report), used by the Umpire to submit the data, and *database* (e.g., Game storage), where the ERIS data are stored. We extend this context with one additional *system asset*—Transmission medium, used to transfer submitted Game report from the *input interface* to the *database*. The example content is illustrated in Fig. 2.3. Security criteria in our example are Confidentiality of the game report and Integrity of the game report.

There might exist several security risks in this context. In this example (see Table 2.2) we present a few risks related to the transmission medium. Let's say there exists a *threat agent*, i.e., an attacker with means to intercept the transmission medium by acting as a proxy between the input interface and the database. Thus, using his expertise and available means, this attacker is able to perform

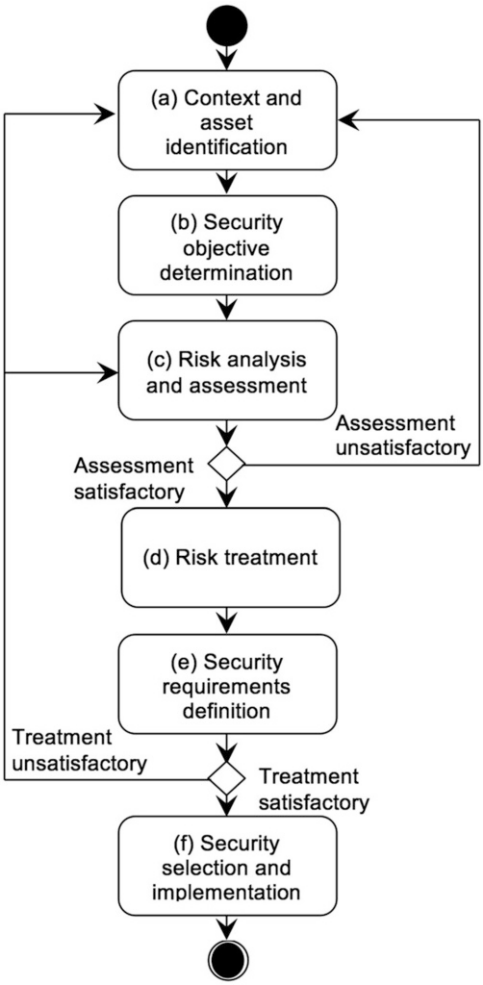

Fig. 2.2 The ISSRM process, adapted from [133]

the *attack method*: firstly, he intercepts the transmission medium between the input interface and the database; secondly, he manipulates data. Data manipulation could be done in two ways (thus characterising two different security risks): (*i*) capturing, modifying and passing data to the database, or (*ii*) capturing, reading, and keeping data for the later use). These security risks are possible because there exists the potential weakness of the transmission medium, which can be intercepted (i.e., *vulnerability* that defines Characteristics of transmission medium to be intercepted) and because there exists no functionality to hide data (i.e., the *vulnerability* that there is a Lack of crypto-functionality at the input interface and database).

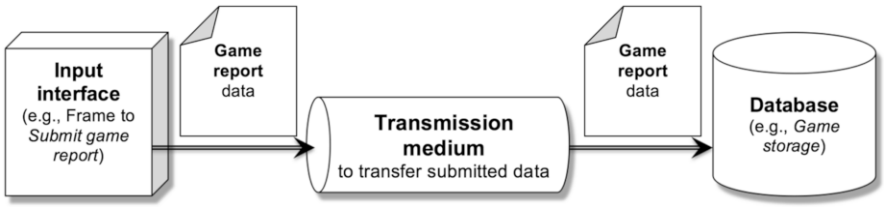

Fig. 2.3 Illustrative example

Table 2.1 Example of the asset-related concepts

Asset-related Concepts		
Assets	**Business assets**	Game report data submitted at the frame to *Submit game report* and stored in the Game storage
	System assets	(*i*) Frame to Submit game report used by umpire, (*ii*)Transmission medium that transfers game report, and (*iii*) Game storage to store ERIS data
Security criteria		(*i*) Confidentiality of game report and (*ii*) Integrity of game report

If the *threat agent* succeeds in executing the *attack method* because of the existing *vulnerabilities*, this will lead to the risk *impact*; i.e., loss of Game report confidentiality or loss of Game report integrity. Additionally, such a risk *event* will also harm both the *IS asset* and the *business asset*: (i) the Game report will not be securely submitted and stored; (ii) the reliability of the transmission medium will be lost.

As compiled in Table 2.2, concepts *threat*, *event*, and *risk* are identified as the composed elements using *threat agent*, *attack method*, *vulnerability*, and *impact*.

We identify (at least) two potential risk treatment decisions (see Table 2.3): risk avoidance and risk reduction. It is important to note that both decisions would mitigate the two previously identified risks (see Table 2.2). In the treatment of *risk avoidance* it is possible to change the transmission medium so that there would be no possibility to intercept it. For example, instead of using the computer means, the user could physically (himself or using postal service) deliver the printed documents; then the receiver would manually enter the Game report in the Game storage.

In the case of the *risk reduction* decision, *security requirements* could be defined to lower the potentiality of a risk event happening. For example, if one makes Game report unreadable to attackers (i.e., cryptography algorithm is implemented as the *security control*), this would mitigate the significance of security risk assessed by the Game report confidentiality. If one defines how to verify the received Game report with the originally sent data (i.e., checksum algorithm is implemented as the *security control*), this would mitigate the significance of security risk assessed by the Game report confidentiality.

Table 2.2 Example of risk-related concepts

Risk-related Concepts		
Risk	An attacker intercepts the transmission medium, and captures, modifies and passes Game report due to the transmission medium characteristic to be intercepted and due to the lack of crypto-functionality at the frame to Submit game report and Game storage, leading to loss of Game report integrity.	An attacker intercepts the transmission medium and captures, reads and keeps the Game report due to the transmission medium characteristic to be intercepted and due to the lack of crypto-functionality at the frame to Submit game report and Game storage, leading to loss of Game report confidentiality.
Impact	• Loss of Game report integrity; • Game report is not securely submitted and stored; • Loss of transmission medium reliability.	• Loss of Game report confidentiality; • Game report is not securely submitted and stored; • Loss of transmission medium reliability.
Event	An attacker intercepts the transmission medium due to its characteristics to be intercepted, captures Game report due to the lack of crypto-functionality at the frame to Submit game report, and modifies and passes data to Game storage due to the lack of crypto-functionality at the Game storage.	An attacker intercepts the transmission medium due to its characteristics to be intercepted, captures Game report due to the lack of crypto-functionality at the frame to Submit game report, and reads and keeps data for later use.
Vulnerability	• Characteristics of transmission medium to be intercepted. • Lack of crypto-functionality at the frame to Submit game report and Game storage.	• Characteristics of transmission medium to be intercepted. • Lack of crypto-functionality at the frame to Submit game report (and Game storage).
Threat	An attacker intercepts the transmission medium, and captures, modifies and passes Game report to Game storage.	An attacker intercepts the transmission medium, and captures, reads and keeps Game report for the later use.
Threat agent	An attacker with means to intercept transmission medium by acting as a proxy.	An attacker with means to intercept transmission medium by acting as a proxy.
Attack method	1. Intercept the transmission medium between the frame to Submit game report and Game storage. 2. Capture, modify and pass data to the Game storage.	1. Intercept the transmission medium between the frame to Submit game report and Game storage. 2. Capture, read and keep Game report for the later use.

Table 2.3 Example of the risk treatment-related concepts

Risk Treatment-related Concepts		
Risk treatment decision	Risk avoidance	Risk reduction
Security requirement	• Change the transmission medium that does not have the ability to be intercepted	• Make Game report unreadable to attackers (mitigates risk to Game report confidentiality) • Verify the received Game report with the original (mitigates risk to Game report integrity)
Control	• Physically delivers the Game report to the football federation. • Game report saved to Game storage by entering it manually	• Cryptographic algorithm • Checksum algorithm

2.6 Further Reading

In general, *security risk management* is an analytical procedure that helps us identify system valuable assets, stakeholders and operations, as well as risk levels of undesirable events. It also provides logic and guidance to find and implement appropriate solutions for specific situations and mitigation strategies. Typically it introduces measures, defined in order to lower the risk level and reduce the likelihood of undesired events. To achieve these goals many different methodologies were developed.

In this section we will survey a few of these security risk management methods. They are useful for understanding the major security risk analysis concepts and principles and how they could be used in various domains. However, this overview is not complete since there exist a large number of approaches[3] to security risk management and different teams might prefer different ways of problem solving.

AURUM (Automated Risk and Utility Management) is a prototype-tool [59] to support decision making according to organisational needs with respect to selection of security measures. AURUM provides different modules regarding the general information security domain. For instance, tools support system characterisation, threat determination, and calculation of risk levels for different organisational assets. The AURUM-supported method consists of a number of steps, where the Bayesian

[3]For example, a recent survey [98] identified *sixteen* security risk management methods.

network method is used to support the activities of risk management. Firstly, it concentrates on the system boundary definition. Secondly, it supports identification of system threats and their sources. The main objective of the third step is to identify managerial, operational and technical vulnerabilities. Next, it introduces controls to mitigate identified risks, to analyse security impact and to support the overall decision making process.

CORAS is a model-driven approach [122], which includes systematic guidance for security risk analysis. The CORAS method includes a language (based on the UML profile and realised by the CORAS tool) and proposes means for documentation, analysis and representation of security risks. The CORAS guidelines include eight steps consisting of a number targeted sub-steps. The main steps are preparation for the analysis, customer presentation of the targets, refining of the target description using asset diagrams, approval of the target description, risk identification using threat diagrams, risk estimation using threat diagrams, risk evaluation using risk diagrams and risk treatment using treatment diagrams. In [170] it is discussed that the CORAS method could be considered as a relevant means to manage cyber-security risks.

CRAMM (CCTA Risk Analysis and Management Method) describes a process-oriented method [212] to analyse risks and to manage those risks through countermeasures. This method is composed of three stages. Firstly, one needs to identify assets and evaluate them. This includes physical assets evaluation from their replacement cost and data and software assets evaluation from the impact of breaches of any of the security objectives (i.e., unavailable, destroyed, disclosed, or modified). Secondly, one needs to assess threats and vulnerabilities from the predefined mappings between threats and assets as well as between threats and their impact. Thirdly, based on the previous steps one needs to define sets of countermeasures that contain necessary information (ranging from high-level security objectives to technical solutions) to manage identified risks.

The *EBIOS* (*French*: Expression des Besoinset Identification des Objectifs de Sécurité) method [50] is used for assessment and treatment of risks relating to information systems security (ISS). It can also be used for communicating this information within the organisation and to its partners. The method's major steps include identification of essential organisation's essential components, determination of security needs and objectives, mapping of security needs to the identified threats, and selection of security requirements to satisfy security needs.

The *MEHARI* (*French*: Methode Harmoniséed' Analyse du Risque Informatique) method [44] is aligned closely with the ISO/IEC 2700x standard family [94, 95]. The method focusses on risk assessment and risk management techniques. It consists of different modules including security stake analysis and classification by identifying and evaluating potential risks and their consequences. The evaluation is guided by security services for assessing the level of system security and by focussing on the main system vulnerabilities. After the risk analysis is performed, the method guides the description of security requirements.

OCTAVE (Operationally Critical Threat, Asset, and Vulnerability Evaluation) is a process-driven approach [8] to identify priorities and manage information security risks. It consists of three stages: first for building asset-based threat profiles from

different levels of organisation; second for identifying the key components and critical assets and evaluating their technological vulnerabilities; and third for risk evaluation, risk profile development and identification of security strategy and risk mitigation plans. There few instantiations of the OCTAVE approach: OCTAVE-S, designed for a small organisation, relies on individuals' knowledge of security and information systems. OCTAVE Allegro proposes a process specifically targeted to the information assets, their usage, storage, transport, and processing, their threats, vulnerabilities, and disruptions.

In this chapter we have presented the ISSRM domain model, its concepts, relationships, metrics and risk management process. In [134] this domain model was developed after analysing security and security risk management standards (see Sect. 1.2), some security modelling and analysis approaches (see Sect. 4.6) and several security risk management methods.

2.7 Exercises

Exercise 2.1 Define what is:

- business asset;
- IS/system asset;
- security criterion;
- threat agent;
- attack method;
- threat;
- vulnerability;
- event;
- impact;
- risk;
- risk treatment (decision);
- security requirement;
- control.

Exercise 2.2 Identify some context in the *Football Federation case*. What are the *business assets* and their *security criteria*? How are these business assets supported by the *system assets*? To structure your answer, fill in Table 2.4.

Table 2.4 Template to support answer of Exercise 2.2

Security context	Football Federation case
Business assets	
System assets	
Security criteria	

Exercise 2.3 For the security context defined in Exercise 2.2, identify at least one *security risk*. Consider the following questions: Who are the *threat agents*? How are the assets *attacked*? What are the *weaknesses/vulnerabilities* of the *system assets*? What is the *impact* of the event on the *system assets* and *business assets*. How does this risk *negate security criteria*? To structure your answer, fill in Table 2.5.

Table 2.5 Template to support answer of Exercise 2.3

Risk constituencies	Football Federation case
Risk	
Impact	
Event	
Vulnerability	
Threat	
Threat agent	
Attack method	

Exercise 2.4 For the security risk identified in Exercise 2.3, what are the possible security *risk treatment decisions*? What are the *security requirements* and how could they be implemented? To structure your answer, fill in Table 2.6.

Table 2.6 Template to support answer of Exercise 2.4

Security risk treatment	Football Federation case
Risk treatment decision	
Security requirements	
Controls	

Exercise 2.5 Following the solutions of Exercises 2.2, 2.3, and 2.4, brainstorm and calculate the *values* of security risk metrics (i.e., business asset value, threat likelihood, vulnerability level, event potentiality, impact level, risk level, security cost, risk reduction level).

Exercise 2.6 How could security risk metrics could be used to perform security trade-off analysis? The answer could be based on examples of risk management given in Sect. 2.5 and elicited in Exercises 2.2–2.5.

Exercise 2.7 Compare different approaches for security risk management to the ISSRM domain model. The comparison could be performed following approach concepts, process steps and changes, assessment metrics and/or other criteria.

References

1. Ahmed, N.: Deriving security requirements from business process models. Ph.D. thesis, University of Tartu, Tartu (2014)
2. Ahmed, N., Matulevičius, R.: Towards transformation guidelines from secure tropos to misuse cases (position paper). In: Proceedings of the SESS'11, pp. 36–42 (2011)
3. Ahmed, N., Matulevičius, R.: A taxonomy for assessing security in business process modelling. In: Proceedings of RCIS 2013, pp. 1–10. IEEE (2013)
4. Ahmed, N., Matulevičius, R.: Securing business process using security risk-oriented patterns. Comput. Stand. Interf. **36**, 723–733 (2014)
5. Ahmed, N., Matulevičius, R., Mouratidis, H.: A model transformation from misuse cases to secure tropos. In: Proceedings of the CAiSE'12 Forum at the 24th International Conference on Advanced Information Systems Engineering (CAiSE), pp. 7–12 (2012)
6. Ahn, G.J., Hu, H.: Towards realizing a formal RBAC model in real systems. In: Proceedings of the 12th ACM Symposium on Access Control Models and Technologies (SACMAT'07), pp. 215–224 (2007)
7. Alalfi, M.H., Cordy, J.R., Dean, T.R.: Recovering role-based access control security models from dynamic web applications. In: Proceedings of the 12th International Conference on Web Engineering (ICWE'12), pp. 121–136 (2012)
8. Alberts, C.J., Dorofee, A.J., Stevens, J., Wooky, C.: Introduction to the OCTAVE approach. Technical Report, Software Engineering Institute, Carnegie Mellon University (2003)
9. Alexander, I.: Initial industrial experience of misuse cases in trade-off analysis. In: Proceedings of the IEEE Joint International Conference on Requirements Engineering, pp. 61–68 (2002)
10. Alexander, I.: Misuse cases: use cases with hostile intent. IEEE Softw. **20**, 58–66 (2003)
11. Alexander, I., Stevens, R.: Writing Better Requirements. Pearson Education Ltd, Boston (2002)
12. Alter, S.: The Work System Method, Connecting People, Processes and IT for Business Results. Work System Press, Larkspur, CA (2006)
13. Altuhhova, O.: An extension of business process model and notation for security risk management. Master thesis, University of Tartu (2013)
14. Altuhhova, O., Matulevičius, R., Ahmed, N.: An extension of business process model and notification for security risk management. Int. J. Inform. Syst. Model. Design (IJISMD) **4**(4), 93–113 (2013)
15. Anderson, R.: Security Engineering: A Guide to Building Dependable Distributed Systems, 2nd edn. Wiley, New York (2008)

R. Matulevičius, *Fundamentals of Secure System Modelling*,
DOI 10.1007/978-3-319-61717-6

16. Apostolopoulos, G., Peris, V., Saha, D.: Transport layer security: how much does it really cost? In: Proceedings IEEE INFOCOM'99 the Conference on Computer Communications, vol. 2, pp. 717–725 (1999)
17. Argyropoulos, N., Alcañiz L.M., Mouratidis, H., Fish, A., Rosado, D.G., de Guzmán, I.G.R., Fernandez-Medina, E.: Eliciting security requirements for business processes of legacy systems. In: Proceedings of PoEM 2015, pp. 91–107 (2015)
18. Asnar, Y., Giorgini, P., Mylopoulos, J.: Goal-driven risk assessment in requirements engineering. Requir. Eng. **16**, 101–116 (2011)
19. Atluri, V., Warner, J.: Security for workflow systems. In: Gertz, M., Jajodia, S. (eds.) Handbook of Database Security, pp. 213–230. Springer, New York (2008)
20. Bandara, A., Shinpei, H., Jurjens, J., Kaiya, H., Kubo, A., Laney, R., Mouratidis, H., Nhlabatsi, A., Nuseibeh, B., Tahara, Y., Tun, T., Washizaki, H., Yoshioka, N., Yu, Y.: Security patterns: comparing modeling approaches. In: Software Engineering for Secure Systems: Industrial and Research Perspectives, pp. 75–111. IGI Global, Hershey, PA (2010)
21. Basin, D., Doser, J., Lodderstedt, T.: Model driven security: From UML models to access control infrastructure. ACM Trans. Softw. Eng. Methodol. (TOSEM) **15**(1), 39–91 (2006)
22. Becker, K.: Pattern and Security Requirements: Engineering-Based Establishment of Security Standards. Springer, New York (2015)
23. Bernardez, B., Duran, A., Genero, M.: Metrics for Use Cases: A Survey of Current Proposals, pp. 59–98. Imperial College Press, London (2005)
24. Bertino, E., Bonatti, P.A., Ferrari, E.: TRBAC: A temporal role-based access control model. ACM Trans. Inf. Syst. Secur. (TISSEC) **4**(3), 191–233 (2001)
25. Blakley, B., Heath, C.: Security design patterns. Technical Report, The Open Group (2004)
26. Börger, E., Cavarra, A., Riccobene, E.: An ASM semantics for UML activity diagrams. In: Proceedings of the 8th AMAST 2000, pp. 293–308. Springer, Berlin (2000)
27. Braun, R., Esswein, W.: Classification of domain-specific BPMN extensions. In: The Practice of Enterprise Modeling. LNBIP, pp. 42–57. Springer, Heidelberg (2014)
28. Braz, F.A., Fernandez, E.B., VanHilst, M.: Eliciting security requirements through misuse cases. In: Proceedings of the 19th International Conference on Database and Expert System Application, pp. 328–333 (2008)
29. Bresciani, P., Perini, A., Giorgini, P., Fausto, G., Mylopoulos, J.: TROPOS: an agent-oriented software development methodology. J. Auton. Agent. Multi-Agent Syst. **25**, 203–236 (2004)
30. Brooke, P.J., Paige, R.F., Power, C.: Approaches to modelling security scenarios with domain-specific languages. In: Security Protocols 2012. LNCS, vol. 7622, pp. 41–54. Springer, Berlin (2012)
31. Brucker, A., Hang, I., Lückemeyer, G., Ruparel, R.: SecureBPMN: modeling and enforcing access control requirements in business processes. In: Proceedings of the 17th ACM Symposium on Access Control Models and Technologies, pp. 123–126. ACM, New York (2012)
32. BSI: BSI Standard 100-1 version 1.5. Information Security Management System (ISMS). Bundesamt für Sicherheit in der Informationstechnik (BSI), Bonn (2008)
33. BSI: BSI Standard 100-2 version 2.0. IT-Grundschutz Methodology. Bundesamt für Sicherheit in der Informationstechnik (BSI), Bonn (2008)
34. BSI: BSI Standard 100-3 version 2.5. Risk Analysis Based on IT-Grundschutz. Bundesamt für Sicherheit in der Informationstechnik (BSI), Bonn (2008)
35. BSI: BSI Standard 100-4 version 1.0. Business Continuity Management. Bundesamt für Sicherheit in der Informationstechnik (BSI), Bonn (2009)
36. CC: Common Criteria for Information Technology Security Evaluation, CC v3.1. Release 4. https://www.commoncriteriaportal.org/cc/ (2015). Last Checked 02 Feb 2016
37. Chang, R.: Defending Against flooding-based distributed denial-of-service attacks: A tutorial. IEEE Commun. Mag. **40**(10), 42–51 (2002)
38. Cherdantseva, Y., Hilton, J., Rana, O.: Towards SecureBPMN: aligning BPMN with the information assurance and security domain. In: Business Process Model and Notation. LNBIP, pp. 107–115. Springer, Heidelberg (2012)

39. Choi, S., Kim, S., Lee, G.: Enhanced misuse case model: a security requirement analysis and specification model. In: Computational Science and Its Applications - ICCSA 2006, pp. 618–625 (2006)
40. Chowdhury, M.J.M., Matulevičius, R., Sindre, G., Karpati, P.: Aligning mal-activity diagrams and security risk management for security requirements definitions. In: Requirements Engineering: Foundation for Software Quality, pp. 132–139. Springer, Heidelberg (2012)
41. Cirit, C., Buzluca, F.: A UML profile for role-based access control. In: Proceedings of the 2nd International Conference on Security of Information and Networks (SIN'09), pp. 83–92 (2009)
42. Clarke, J.: SQL Injection Attacks and Defense. Syngress Publishing, Boston (2011)
43. Clavel, M., Silva, V., Braga, C., Egea, M.: Model-driven security in practice: an industrial experience. In: Proceedings of the 4th European Conference on Model Driven Architecture: Foundations and Applications (ECMDA-FA'06), pp. 326–337. Springer, Berlin (2008)
44. CLUSIF: MEHARI 2010: Fundamental concepts and principles-specifications. Technical Report, Club de la Securite de L'Information Francais (2010)
45. Cockburn, A.: Writing Effective Use Cases. Addison-Wesley, Boston (2001)
46. Dalpiaz, F., Paja, E., Giorgini, P.: Security Requirements Engineering: Designing Secure Socio-Technical Systems. The MIT Press, Cambridge (2016)
47. Damiani, M.L., Bertino, E., Catania, B., Perlasca, P.: Geo-RBAC: A spatially aware RBAC. ACM Trans. Inform. Syst. Secur. (TISSEC) **10**, 1 (2007)
48. Danezis, G., Domingo-Ferrer, J., Hansen, M., Hoepman, J.H., Le Métayer, D., Tirtea, R., Schiffner, S.: Privacy and data protection by design – from policy to engineering. Technical Report, ENISA (2014)
49. Davis, A., Overmeyer, S., Jordan, K., Caruso, J., Dandashi, F., Dinh, A., Kincaid, G., Reynolds, P., Sitaram, P., Ta, A., Theofanos, M.: Identifying and measuring quality in a software requirements specification. In: Proceedings of the 1st International Software Metrics Symposium, pp. 141–152 (1993)
50. DCSSI Advisory Office: EBIOS 2010: Expression of Needs and Identification of Security Objectives. Technical Report, Secrétariat général de la défense nationale, Direction centrale de la sécurité des systèmes d'information (2010)
51. De Win, B., Scandariato, R., Buyens, K., Grégoire, J., Joosen, W.: On the secure software development process: CLASP, SDL and Touchpoints compared. Inf. Softw. Technol. **51**, 1152–1171 (2009)
52. Dougherty, C., Sayre, K., Seacord, R.C., Svoboda, D., Togashi, K.: Secure Design Patterns. Technical Report, Software Engineering Institute (2009)
53. Dubois, E., Heymans, P., Mayer, N., Matulevičius, R.: A Systematic Approach to Define the Domain of Information System Security Risk Management, pp. 289–306. Springer, New York (2010)
54. Dumas, M., La Rosa, M., Mendling, J., Reijers, H.: Fundamentals of Business Process Management. Springer, New York (2013)
55. Dwork, C.: Differential privacy. In: Automata, Languages and Programming, pp. 1–12. Springer, New York (2006)
56. Dwork, C.: Differential privacy: a survey of results. In: Theory and Application of Models in Computation, pp. 1–19. Springer, New York (2008)
57. Dwork, C.: Differential privacy in new settings. In: Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 174–183. Society for Industrial and Applied Mathematics, Philadelphia (2010)
58. Easterbrook, S.: Fundamentals of requirements engineering. http://www.cs.toronto.edu/~sme/CSC340F/ (2004). Last Checked 30 May 2016
59. Ekelhart, A., Fenz, S., Neubauer, T.: AURUM: a framework for information security risk management. In: Proceedings of the 42nd Hawaii International Conference on System Sciences (2009)
60. El-Attar, M.: From misuse cases to mal-activity diagrams: Bridging the gap between functional security analysis and design. Softw. Syst. Model. **13**, 173–190 (2014)

61. El-Hadary, H., El-Kassas, S.: Capturing security requirements for software systems. Cairo Univ. J. Adv. Res. **5**, 463–472 (2014)
62. Elahi, G., Yu, E.: A goal oriented approach for modeling and analyzing security trade-offs. In: Parent, C., Schewe, K.D., Storey, V.C., Thalheim, B. (eds.) Proceedings of the 26th International Conference on Conceptual Modelling (ER 2007), vol. 4801, pp. 87–101. Springer, Berlin (2007)
63. Fabian, B., Gürses, S., Heisel, M., Santen, T., Schmidt, H.: A comparison of security requirements engineering methods. Requir. Eng. **15**(1), 7–40 (2010)
64. Fernandez-Buglioni, E.: Security Patterns in Practice: Designing Secure Architectures Using Software Patterns. Wiley, New York (2013)
65. Ferraiolo, D., Sandhu, R., Gavrila, S., Kuhn, D.R., Chandramouli, R.: Proposed NIST standard for role-based access control. ACM Trans. Inf. Syst. Secur. **4**(3), 224–274 (2001)
66. Firesmith, D.: Common concepts underlying safety, security and survivability engineering. Technical Report, CMU/SEI-2003-TN-033, Software Engineering Institute (2003)
67. Firesmith, D.: Security use cases. J. Object Technol. **2**(3), 53–64 (2003)
68. Firesmith, D.G.: Engineering security requirements. J. Object Technol. **2**(1), 53–68 (2003)
69. Firesmith, D.: A taxonomy of security-related requirements. In: Proceedings of the Fourth International Workshop on Requirements Engineering for High-Availability Systems (RHAS'05), p. 11 (2005)
70. Firesmith, D.: Engineering safety- and security-related requirements for software- intensive systems. Tutorial, 2007 Carnegie Mellon University (2007)
71. Firesmith, D.: Engineering safety and security related requirements for software intensive systems. In: 29th International Conference on Software Engineering: Companion, p. 169. IEEE Computer Society (2007)
72. Fogie, S., Grossman, J., Hansen, R., Rager, A., Petkov, P.D.: XSS Attacks: Cross Site Scripting Exploits and Defense. Syngress Publishing, Burlington, MA (2007)
73. Fung, B.C.M., Wang, K., Chen, R., Yu, P.S.: Privacy-preserving data publishing: a survey of recent developments. ACM Comput. Surv. **42**(4), 1–53 (2010)
74. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, Boston (1994)
75. Garcia-Alfaro, J., Navarro-Arribas, G.: A survey on detection techniques to prevent cross-site scripting attacks on current web applications. In: Critical Information Infrastructures Security, pp. 287–298. Springer, Berlin (2008)
76. Gharib, M., Giorgini, P., Mylopoulos, J.: Ontologies for privacy requirements engineering: a systematic literature review. Technical Report, University of Florence and University of Trento (2016)
77. Giorgini, P., Massacci, F., Mylopoulos, J., Zannone, N.: Modeling security requirements through ownership, permission and delegation. In: Proceedings of the 13th IEEE International Conference on Requirements Engineering (RE'05). IEEE Computer Society (2005)
78. Giorgini, P., Massacci, F., Mylopoulos, J., Zannone, N.: Modelling social and individual trust in requirements engineering methodologies. In: Proceedings of the 3rd International Conference on Trust Management. LNCS, pp. 161–176. Springer, Heidelberg (2005)
79. Goldkuhl, G., Lind, M., Seigerroth, U.: Method integration: the need for a learning perspective. IEEE Proc. Softw. (Special issue on Information System Methodologies) **145**(4), 113–118 (1998). http://ieeexplore.ieee.org/document/729576/
80. Goldstein, A., Frank, U.: Components of a multi-perspective modeling method for designing and managing IT security systems. Inf. Syst. E-Bus Manag. **14**, 101–140 (2016)
81. Gopalakrishnan, S., Krogstie, J., Sindre, G.: Extending use and misuse case diagrams to capture multi-channel information systems. In: Informatics Engineering and Information Science, pp. 355–369 (2011)
82. Graham, D.: Introduction to the CLASP Process. Build Security In (2006)
83. Haibo, S., Fan, H.: A context-aware role-Based access control model for web services. In: IEEE International Conference on E-Business Engineering, pp. 220–223 (2005)

84. Haley, C., Laney, R., Moffet, J.D., Nuseibeh, B.: Security requirements engineering: a framework for representation and analysis. IEEE Trans. Softw. Eng. **34**(1), 133–153 (2008)
85. Hansen, F., Oleshchuk, V.: SRBAC: A spatial role-based access control model for mobile systems. In: Proceedings of the 7th Nordic Workshop on Secure IT Systems (NORDSEC03), pp. 129–141 (2003)
86. Hartong, M., Goel, R., Wijesekera, D.: UsemMisuse case driven analysis of positive train control. In: Advances in Digital Forensics II, pp. 141–155. Springer, Berlin (2006)
87. Herzberg, A., Jarecki, S., Krawczyk, H., Yung, M.: Proactive secret sharing or: how to cope with perpetual leakage. In: Proceedings of the CRYPTO'95, pp. 339–352 (1995)
88. Howard, M., Lipner, S.: The Security Development Lifecycle. Microsoft Press, Remond (2006)
89. Hu, V.C., Ferraiolo, D., Kuhn, R., Schnitzer, A., Sandlin, K., Miller, R., Scarfone, K.: Guide to attribute based access control (ABAC) definition and considerations. Technical Report 800–162, NIST Special Publication (2014)
90. Hu, V.C., Kuhn, D.R., Ferraiolo, D.F.: Attribute-based access control. Computer **2**, 85–88 (2015)
91. Hummer, W., Gaubatz, P., Strembeck, M., Zdun, U., Dustdar, S.: Enforcement of entailment constraints in distributed service-based business processes. Inf. Softw. Technol. **55**(11), 1884–1903 (2013)
92. Hussein, M., Zulkernine, M.: Intrusion detection aware component-based systems: a specification-based framework. J. Syst. Softw. **80**(5), 700–710 (2007)
93. ISO/IEC: 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 (2004)
94. ISO/IEC: ISO/IEC 27005: 2011: Information Technology: Security Techniques: Information Security Risk Management. International Organization for Standardization, Geneva (2011)
95. ISO/IEC: ISO/IEC 27001:2013: Information Technology – Security Techniques – Information Security Management Systems – Requirements. International Organization for Standardization, Geneva (2013)
96. Jackson, M.: Problem Frames: Analysing and Structuring Software Development Problems. Addison-Wesley, Boston (2001)
97. Jaks, L.: A prototype for transforming role-based access control models. Bachelor thesis, University of Tartu (2012)
98. Janulevičius, J.: Method of information security risk analysis for virtualized systems. Ph.D. thesis, Vilnius Gediminas Technical University (2016)
99. Jensen, J., Tøndel, I.A., Meland, P.M.: Experimental threat model reuse with misuse case diagrams. In: Information and Communication Security, pp. 355–366 (2010)
100. Jürjens, J.: Secure System Development with UML. Springer, Berlin (2005)
101. Karpati, P., Sindre, G., Opdahl, A.L.: Visualizing cyber attacks with misuse case maps. In: Wieringa, R., Persson, A. (eds.) Proceedings of the Requirements Engineering: Foundation for Software Quality (REFSQ 2010). Springer, Heidelberg (2010)
102. Karpati, P., Opdahl, A.L., Sindre, G.: Experimental comparison of misuse case maps with misuse cases and system architecture diagrams for eliciting security vulnerabilities and mitigations. In: Proceedings of the 6th International Conference on Availability, Reliability, and Security (2011)
103. Karpati, P., Sindre, G., Matulevičius, R.: Comparing misuse case and mal-activity diagrams for modelling social engineering attacks. Int. J. Secure Softw. Eng. **3**(2), 54–73 (2012)
104. Karpati, P., Redda, Y., Opdahl, A.L., Sindre, G.: Comparing attack trees and misuse cases in an industrial setting. Inf. Softw. Technol. **56**, 294–308 (2014)
105. Kim, S., Narasimha Reddy, A.L.: Statistical techniques for detecting traffic anomalies through packet header data. IEEE/ACM Trans. Netw. **16**(3), 562–575 (2008)
106. Kissel, R.: Glossary of key information security terms. Technical Report NISTIR 7298 revision 2, NIST (2013)

107. Kolk, K.: An empirical comparison of approaches for security requirements elicitation. Master thesis, University of Tartu (2015)
108. Korman, M., Lagerström, R., Ekstedt, M.: Modeling authorization in enterprise-wide contexts. In: PoEM-SDC 2015: Short and Doctoral Consortium Papers at PoEM 2015, pp. 81–90 (2015)
109. Krogstie, J.: Model-Based Development and Evolution of Information Systems. Springer, London (2012)
110. Kulak, D., Guiney, E.: Use Cases: Requirements in Context, 2nd edn. Addison-Wesley, New York (2004)
111. Lakk, H.: Model-driven role-based access control for databases. Master thesis, University of Tartu (2012)
112. van Lamsweerde, A.: Elaborating security requirements by construction of intentional anti-models. In: Proceedings of 26th International Conference on Software Engineering (ICSE'04), pp. 148–157 (2004)
113. van Lamsweerde, A.: Requirements Engineering: From System Goals to UML Models to Software Specifications. Wiley, New York (2009)
114. Lee, S.W., Gandhi, R., Muthurajan, D., Yavagal, D., Ahn, G.J.: Building problem domain ontology from security requirements in regulatory documents. In: Proceedings of the 2006 International Workshop on Software Engineering for Secure Systems (2006)
115. Leoni, D.: Non-interactive differential privacy: a survey. In: Proceedings of the 1st International Workshop on Open Data (WOD'12), pp. 40–52 (2012)
116. Li, T., Horkoff, J.: Dealing with security requirements for socio-technical systems: a holistic approach. In: Proceedings of International Conference on Advanced Information Systems Engineering (CAiSE 2014), pp. 285–300. Springer, Heidelberg (2014)
117. Lin, L., Nuseibeh, B., Ince, D., Jackson, M.: Using abuse frames to bound the scope of security problem. In: Proceedings of the 12th IEEE International Requirements Engineering Conference (2004)
118. Liu, L., Yu, E., Mylopoulos, J.: Security and privacy requirements analysis within a social setting. In: Proceedings of the 11th IEEE International Requirements Engineering Conference (RE'03), p. 151. IEEE Computer Society (2003)
119. Lodderstedt, T., Basin, D., Doser, J.: SecureUML: a UML-based modeling language for model-driven security. In: Proceedings of the 5th International Conference on the Unified Modeling Language, vol. 2460, pp. 426–441. Springer, Berlin (2002)
120. Lord, N.: Common Malware Types: Cybersecurity 101. https://www.veracode.com/blog/2012/10/common-malware-types-cybersecurity-101/ (2012)
121. Loukas, G., Öke, G.: Protection against denial of service attacks. Comput. J. **53**(7), 1020–1037 (2010)
122. Lund, M.S., Solhaug, B., Stølen, K.: Model-Driven Risk Analysis: The CORAS Approach. Springer, Heidelberg (2011)
123. Mahajan, R., Bellovin, S.M., Floyd, S., Ioannidis, J., Paxson, V., S., S.: Controlling high bandwidth aggregates in the network. SIGCOMM Comput. Commun. Rev. **32**(3), 62–73 (2002)
124. Marcinkowski, B., Kuciapski, M.: A business process modeling notation extension for risk handling. In: 11th International Conference on Information Systems and Industrial Management. LNCS, pp. 374–381. Springer, Heidelberg (2012)
125. Massaci, F., Mylopoulos, J., Zannone, N.: Security requirements engineering: The SI* modelling language and the tropos methodology. Adv. Intell. Inf. Syst. **265**, 147–174 (2010)
126. Matulevičius, R.: Comparing modelling languages for information systems security risk management. In: Seyff, N., Koziolek, A. (eds.) Modellin and Quality in Requirements Engineering: Essays Dedicated to Martin Glinz on the Occasion of His 60th Birthday, pp. 207–220. Monsenstein and Vannerdat (2012)
127. Matulevičius, R., Dumas, M.: A comparison of SecureUML and UMLsec for role-based access control. In: The 9th Conference on Databases and Information Systems (Baltic DB and IS 2010), pp. 171–185 (2010)

128. Matulevičius, R., Dumas, M.: Towards model transformation between SecureUML and UMLsec for role-based access control. In: Databases and Information Systems VI, pp. 339–352. IOS Press, Amsterdam (2011)
129. Matulevičius, R., Lakk, H.: A model-driven role-based access control for SQL databases. Complex Syst. Inform. Model. Q. **3**, 35–62 (2015)
130. Matulevičius, R., Mayer, N., Heymans, P.: Alignment of misuse cases with security risk management. In: Proceedings of the ARES 2008 Symposium on Requirements Engineering for Information Security (SREIS 2008), pp. 1397–1404. IEEE Computer Society (2008)
131. Matulevičius, R., Mayer, N., Mouratidis, H., Dubois, E., Heymans, P., Genon, N.: 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 System Engineering (CAiSE 2008). Springer, Berlin/Heidelberg (2008)
132. Matulevičius, R., Mayer, N., Mouratidis, H., Dubois, E., Heymans, P.: Syntactic and semantic extensions to secure tropos to support security risk management. J. UCS **18**(6), 816–844 (2012)
133. Mayer, N.: Model-based management of information system security risk. Ph.D. thesis, University of Namur (2009)
134. Mayer, N., Heymans, P., Matulevičius, R.: Design of a modelling language for information system security risk management. In: Proceedings of the Research Challenges in Information Science (RCIS 2007), pp. 121–131 (2007)
135. Mayer, N., Dubois, E., Matulevičius, R., Heymans, P.: Towards a measurement framework for security risk management. In: Proceedings of the Workshop on Modeling Security (MODSEC08) held as part of MODELS 2008 (2008)
136. McDermott, J., Fox, C.: Using abuse case models for security requirements analysis. In: Proceedings of the 15th Annual Computer Security Applications Conference (ACSAC '99) (1999)
137. McGraw, G.: Software Security: Building Security In. Addison-Wesley, Upper Saddle River (2006)
138. McGraw, R.W.: Risk-adaptable access control (RAdAC). In: Privilege (Access) Management Workshop. NIST–National Institute of Standards and Technology – Information Technology Laboratory (2009)
139. Mead, N.R., Stehney, T.: Security quality requirements engineering (SQUARE) methodology. In: Software Engineering for Secure Systems (SESS05) (2005)
140. Mead, N.R., Hough, E.D., Stehney II, T.R.: Security quality requirements engineering (SQUARE) methodology. Technical Report CMU/SEI-2005-TR-009, ESC-TR-2005-009, Software Engineering Institute (2005)
141. Mellado, D., Fernández-Medina, E., Piattini, M.: A common criteria based security requirements engineering process for the development of secure information systems. Comput. Stand. Interf. **29**, 244–253 (2007)
142. Mellado, D., Fernández-Medina, E., Piattini, M.: Towards security requirements management for software product lines: a security domain requirements engineering process. Comput. Stand. Interf. **30**, 361–371 (2008)
143. Mellado, D., Blanco, C., Sánchez, L.E., Fernández-Medina, E.: A systematic review of security requirements engineering. Comput. Stand. Interf. **32**, 153–165 (2010)
144. Menzel, M., Thomas, I., Meinel, C.: Security requirements specification in service-oriented business process management. In: International Conference on Availability, Reliability and Security (ARES 2009), pp. 41–49 (2009)
145. MITRE: Common Attack Pattern Enumeration and Classification. https://capec.mitre.org
146. Moffett, J., Nuseibeh, B.: A framework for security requirements engineering. Technical Report 368, Department of Computer Science, University of York (2003)
147. Mouratidis, H.: 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. Ph.D. thesis, Department of Computer Science, University of Sheffield, UK (2004)

148. Mouratidis, H., Giorgini, P.: Secure tropos: a security-oriented extension of the tropos methodology. Int. J. Softw. Eng. Knowl. Eng. (IJSEKE) **17**(2), 285–309 (2007)
149. Mouratidis, H., Jurjens, J.: From goal-driven security requirements engineering to secure design. Int. J. Intell. Syst. **25**, 813–840 (2010)
150. Mouratidis, H., Giorgini, P., Manson, G.: Integrating security and systems engineering: towards the modelling of Secure information systems. In: Proceedings of the 15th Conference on Advanced Information Systems Engineering (CAiSE'03), pp. 63–78. Springer, Berlin (2003)
151. Mülle, J., Stackelberg, S., Bohm, K.: A security language for BPMN process models. Technical Report 9, Karlsruhe Reports in Informatics (2011)
152. Myagmar, S., Lee, A.J., Yurcik, W.: Threat modeling as a basis for security requirements. In: SREIS (2005)
153. Natan, R.B.: Implementing Database Security and Auditing: Includes Examples for Oracle, SQL Server, DB2 UDB, Sybase. Digital Press, Clifton (2005)
154. NIST: NIST Special Publication 800-39. Managing Information Security Risk: Organization, Mission, and Information System View. National Institute of Standards and Technology, Gaithersburg (2011)
155. NIST: NIST Special Publication 800-30. Guide for Conducting Risk Assessments. National Institute of Standards and Technology, Gaithersburg (2012)
156. Noh, S., Lee, C., Choi, K., Jung, G.: Detecting distributed denial of service (DDoS) attacks through inductive learning. In: Intelligent Data Engineering and Automated Learning, pp. 286–295. Springer, Berlin (2003)
157. Nuseibeh, B., Finkelstein, A., Kramer, J.: Method engineering for multi-perspective software development. Inf. Softw. Technol. J. **38**(4), 267–274 (1996). http://www.sciencedirect.com/science/article/pii/0950584995010548
158. OMG: Unified Modeling Language: Infrastructure and Superstructure, version 2.0. http://www.omg.org/spec/UML/2.0/ (2005)
159. OMG: Business Process Model and Notation (BPMN), version 2.0. http://www.omg.org/spec/BPMN/2.0/ (2011)
160. Onchukova, A.: Security risk management using misuse cases and mal-activities. Master thesis, University of Tartu (2013)
161. Opdahl, A.L., Henderson-Sellers, B.: A unified modelling language without referential redundancy. Data Knowl. Eng. (DKE) (Special Issue on Quality in Conceptual Modelling) **55**, 277–300 (2005)
162. Opdahl, A.L., Sindre, G.: Experimental comparison of attack trees and misuse cases for security threat identification. Inf. Softw. Technol. **51**, 916–932 (2009)
163. ben Othmane, L., Ranchal, R., Fernando, R., Bhargava, B., Bodden, E.: Incorporating attacker capabilities in risk estimation and mitigation. Comput. Secur. **51**, 41–61 (2014)
164. Park, J., Sandhu, R.: The UCON ABC usage control model. ACM Trans. Inf Syst. Secur. (TISSEC) **7**(1), 128–174 (2004)
165. Pauli, J.J., Xu, D.: Misuse case-based design and analysis of secure software architecture. In: Proceedings of the International Conference in Information Technology: Coding and Computing (ITCC'05), pp. 398–403 (2005)
166. Pavlidis, M., Islam, S.: SecTro: a CASE tool for modelling security in requirements engineering using secure tropos. In: Proceedings of the CAiSE Forum (2011)
167. Peeters, J.: Agile security requirements engineering. In: SREIS (2005)
168. Pfitzmann, A., Hansen, M.: A terminology for talking about privacy by data minimization: anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management. Technical Report, TU Dresden and ULD Kiel (2010)
169. Radhakrishnan, R.: The Fifth and Final Frontier of Access Control Model (2012). http://www.isaca-washdc.org/presentations/2012/201211-session3_article.pdf
170. Refsdal, A., Solhaug, B., Stølen, K.: Cyber-Risk Management. Springer, Cham (2015)
171. Rodriguez, A., Fernandez-Medina, E., Piattini, M.: A BPMN extension for the modeling of security requirements in business processes. IEICE Trans. Inf. Syst. **90**(4), 745–752 (2007)

172. Rodríguez, A., de Guzmán, I.G.R., Fernández-Medina, E., Piattini, M.: Semi-formal transformation of secure business processes into analysis class and use case models: an MDA approach. Inf. Softw. Technol. **52**, 945–971 (2010)
173. Rodríguez, A., Fernández-Medina, E., Trujillo, J., Piattini, M.: Secure business process model specification through a UML 2.0 activity diagram profile. Decis. Support. Syst. **51**, 446–465 (2011)
174. Rostad, L.: An extended misuse case notation: including vulnerabilities and the insider threat. In: Proceedings of the 12th Working Conference REFSQ'06 (2006)
175. Rrenja, A., Matulevičius, R.: Pattern-based security requirements derivation from secure tropos models. In: Proceedings of PoEM 2015, pp. 59–74 (2015)
176. Sánchez, P., Moreira, A., Fuentes, L., Araújo, J., Magno, J.: Model-driven development for early aspects. Inf. Softw. Technol. **52**, 249–273 (2010)
177. Sandhu, R.S., Coyne, E.J., Feinstein, H.L., Youman, C.E.: Role-based access control models. IEEE Comput. **29**(2), 38–47 (1996)
178. Sandkuhl, K., Matulevičius, R., Ahmed, N., Kirikova, M.: Refining security requirement elicitation from business process using method engineering. In: Joint Proceedings of the BIR 2015 Workshops and Doctoral Consortium (2015)
179. Scandariato, R., Wuyts, K., Joosen, W.: A descriptive study of Microsoft's threat modeling technique. Requir. Eng. J. **20**, 163–180 (2015)
180. Schleicher, D., Leymann, F., Schumm, D., Weidmann, M.: Compliance scopes: Extending the BPMN 2.0 Meta model to specify compliance requirements. In: IEEE International Conference on Service-Oriented Computing and Applications (SOCA), pp. 1–8. IEEE (2010)
181. Schneier, B.: Attack trees. Dr. Dobb's J. (1999). https://www.schneier.com/academic/archives/1999/12/attack_trees.html
182. Schumacher, M., Fernandez-Buglioni, E., Hybertson, D., Buschmann, F., Sommerlad, P.: Security Patterns: Integrating Security and System Engineering. Wiley, New York (2005)
183. Sergeev, A.: Role based access control as secureUML model in web applications development with spring security. Master thesis, University of Tartu (2016)
184. Shafiq, B., Samuel, A., Ghafoor, H.: A GTRRBAC based system for dynamic workflow composition and management. In: Eighth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC), pp. 284–290 (2005)
185. Shaikh, R.A., Adi, K., Logrippo, L.: Dynamic risk-based decision methods for access control systems. Comput. Secur. **31**(4), 447–464 (2012)
186. Shen, Y., Pearson, S.: Privacy enhancing technologies: a review. Technical Report HPL-2011-113, HP Laboratories (2011)
187. Shin, M.E., Gomaa, H.: Software requirements and architecture modeling for evolving non-secure applications into secure applications. Sci. Comput. Program **66**(1), 60–70 (2007)
188. Shostack, A.: Threat Modeling: Designing for Security. Wiley, New York (2014)
189. Silver, B.: BPMN Method and Style: A Levels-Based Methodology for BPMN Process Modeling and Improvement Using BPMN 2.0. Cody-Cassidy Press, Aptos (2009)
190. Sindre, G.: A look at misuse cases for safety concerns. In: Situational Method Engineering: Fundamentals and Experiences, pp. 252–266. Springer, Boston (2007)
191. Sindre, G.: Mal-activity diagrams for capturing attacks on business processes. In: Requirements Engineering: Foundation for Software Quality, pp. 355–366. Springer, Heidelberg (2007)
192. Sindre, G., Opdahl, A.L.: Eliciting security requirements with misuse cases. Requir. Eng. J. **10**(1), 34–44 (2005)
193. Sing, E.: A prototype to transform models of secure tropos and misuse case diagrams. Bachelor thesis, University of Tartu (2014)
194. Slavin, R., Lehker, J.M., Niu, J., Breaux, T.: Managing security requirements patterns using feature diagram hierarchies. In: Proceedings of RE 2014, pp. 193–202 (2014)
195. Sommestad, T., Ekstedt, M., Holm, H.: The cyber security modeling language: a tool for assessing the vulnerability of enterprise system architectures. IEEE Syst. J. **7**(3), 363–373 (2013)

196. Soomro, I., Ahmed, N.: Towards security risk-oriented misuse cases. In: Business Process Management Workshops. LNBIP, pp. 689–700. Springer, Heidelberg (2012)
197. Staron, M.: Adopting model driven software development in industry – a case at two companies. In: Model Driven Engineering Languages and Systems, pp. 57–72 (2006)
198. Stålhane, T., Sindre, G.: Safety hazard identification by misuse cases: Experimental comparison of text and diagrams. In: Model Driven Engineering Languages and Systems, pp. 721–735 (2008)
199. Tark, K.: Role based access model in XML based documents. Master thesis, University of Tartu (2013)
200. Tark, K., Matulevičius, R.: Short paper: role-based access control for securing dynamically created documents. In: Business Process Management Workshops. LNBIP, vol. 171, pp. 520–525. Springer, Berlin (2014)
201. Thomas, R., Mark, B., Johnson, T., Croall, J.: NetBouncer: Client-legitimacy-based high-performance DDoS filtering. In: DARPA Information Survivability Conference and Exposition, vol. 12003, pp. 14–25 (2003)
202. Tsipenyuk, K., Chess, B., McGraw, G.: Seven pernicious kingdoms: a taxonomy of software security errors. IEEE Secur. Priv. **3**(4), 81–84 (2005). http://ieeexplore.ieee.org/document/1556543/
203. Tsoumas, B., Gritzalis, D.: Towards an ontology-based security management. In: Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA'06), vol. 1 (2006)
204. Tsoumas, B., Papagiannakopoulos, P., Dritsas, S., Gritzalis, D.: Security-by-ontology: a knowledge-centric approach. In: Boston, S. (ed.) Security and Privacy in Dynamic Environments, pp. 99–110. Springer, London (2006)
205. Uzunov, A.V., Fernandez, E.B.: An extensible pattern-based library and taxonomy of security threats for distributed systems. Comput. Stand. Interf. **36**(4), 734–747 (2014)
206. Viega, J.: Building security requirements with CLASP. In: Proceedings of the 2005 Workshop on Software Engineering for Secure Systems—Building Trustworthy Applications, pp. 1–7 (2005)
207. Wand, Y., Weber, R.: On the ontological expressiveness of information systems analysis and design grammars. J. Inf. Syst. **3**, 217–237 (1993)
208. Whittle, J., Wijesekera, D., Hartong, M.: Executable misuse cases for modeling security concerns. In: Proceedings of the 30th International Conference on Software Engineering (ICSE'08), pp. 121–130 (2008)
209. Withall, S.: Software Requirements Patterns. Microsoft Press, Sebastopol (2007)
210. Xin, J.: Applying model driven architecture approach to model role based access control system. Master thesis, University of Ottawa (2006)
211. Xu, D., Pauli, J.: Threat-driven design and analysis of secure software architectures. J. Inf. Assur. Secur. **1**(3), 171–180 (2006)
212. Yazar, Z.: A qualitative risk analysis and management tool: CRAMM. Technical Report, SANS Institute (2002)
213. Yu, E.: Towards modeling and reasoning support for early-phase requirements engineering. In: Proceedings of the 3rd IEEE International Symposium on Requirements Engineering (RE'97), p. 226. IEEE Computer Society (1997)
214. Yue, T., Briand, L.C., Labiche, Y.: A systematic review of transformation approaches between user requirements and analysis models. Requir. Eng. J. **16**, 75–99 (2011)
215. Zuccato, A.: Holistic security requirement engineering for electronic commerce. Comput. Secur. **23**, 63–76 (2004)
216. Zuccato, A.: Holistic security management framework applied in electronic commerce. Comput. Secur. **26**(3), 256–265 (2007)
217. Zuccato, A., Endersz, V., Daniels, N.: Security requirements engineering at a telecom provider. In: ARES, pp. 1139–1147 (2008)