Towards Definition of Secure Business Processes

Olga Altuhhova, Raimundas Matulevičius, and Naved Ahmed

Institute of Computer Science, University of Tartu
J. Liivi 2, 50409 Tartu, Estonia
olgaaltuhhova@hotmail.com, {rma,naved}@ut.ee

Abstract. Business process modelling is one of the major aspects in the modern system development. Recently business process model and notation (BPMN) has become a standard technique to support this activity. Although BPMN is a good approach to understand business processes, there is a limited work to understand how it could deal with business security and security risk management. This is a problem, since both business processes and security concerns should be understood in parallel to support a development of the secure systems. In this paper we analyse BPMN with respect to the domain model of the IS security risk management (ISSRM). We apply a structured approach to understand key aspects of BPMN and how modeller could express secure assets, risks and risk treatment using BPMN. We align the main BPMN constructs with the key concepts of the ISSRM domain model. We show applicability of our approach on a running example related to the Internet store. Our proposal would allow system analysts to understand how to develop security requirements to secure important assets defined through business processes. In addition we open a possibility for the business and security model interoperability and the model transformation between several modelling approaches (if these both are aligned to the ISSRM domain model).

Keywords: Business process model and notation (BPMN), Security risk management, Alignment of modelling languages, Information systems.

1 Introduction

Business process modelling takes an important part when developing information systems (IS). It helps specify standard and optimised workflows of organisation. The business processes that involve many participants, their communications, necessary resources and their usage not only extend organisational competiveness but also increase business vulnerabilities. Thus, understanding and modelling of IS security becomes an important activity during IS development. Security refers to the capability of a product, i.e., IS, to protect data and information against the unauthorised access by persons or systems that have intention to harm it.

Identification of the security requirements is typically performed only after the business process has been defined. Furthermore, it is observed [12] that security considerations often arise most usually during *implementation* or *maintenance* stages. Firstly, this means that security engineers get little feedback about the need for system security. Secondly, security risks are very hard to calculate: security-critical systems

M. Bajec and J. Eder (Eds.): CAiSE 2012 Workshops, LNBIP 112, pp. 1–15, 2012.

are characterised by the fact that the occurrence of a successful attack at one point in time on a given system increases the likelihood that the attack will be launched subsequently at another system point. This is a serious hindrance to secure system development, since the early consideration of security (e.g., when defining the business processes) allows engineers to envisage threats, their consequences and design countermeasures. Then the system design and architecture alternatives, that do not offer a sufficient security level, could be discarded.

Although there exists few attempts to introduce notations to address security at the business process modelling (i.e., [16] [19] [20]) or to relate business process and security requirements modelling (i.e., [17]), these are rather at the coarse-grained level. In principle, the approaches do not illustrate guidelines on how to advance from one security aspect to another, or how to understand security concerns and define security requirements.

In this work we consider Business Process Model and Notation (BPMN, version 2.0) [18] [21], a multi-vendor standard controlled by the Object Management Group [24]. The primary purpose of BPMN is modelling of the business processes. Like in other modelling languages, BPMN notations are linked to a semantic model, which means that each shape has a specific meaning, and defined rules to connect objects. In this work our goal is *not* to develop new modelling approach for security, but rather to understand (*i*) how business activities expressed using BPMN could be annotated with the security concerns; (*ii*) how BPMN could be used to define security requirements; and (*iii*) how the BPMN language itself could be used to reason for the security requirements through illustration of the potential security risks. In this paper we specifically address the second (*ii*) and third (*iii*) aspect.

To achieve our goal we have selected a domain model [7] [15] for IS security risk management (ISSRM) and have aligned the BPMN constructs to the concepts of this domain model. We result in a grounded and fine-grained reasoning for extensions of BPMN toward secure business processes. In addition we present our analysis through an illustrative example; thus, in this way we end up with guidelines for the BPMN application to analyse security risks.

The paper structure is as follows: in Section 2 we give the background to our study. Based on the running example in Section 3 we investigate BPMN following the ISSRM process. In Section 4 we present an alignment of BPMN constructs to the concepts of ISSRM. In Section 5 we discuss our finding, related work and conclude our study.

2 Background

2.1 Security Analysis Methods

To model secure systems, different security risk management approaches are developed. For instance, CORAS is a model-driven approach [4], which includes a systematic guidance for security risk analysis. The Tropos Goal-Risk framework [2] supports modelling, assessing and treating risks on the basis of the likelihood and severity of failures. This framework consists of three conceptual layers – strategy, event, and treatment to assess the risk of some events and evaluate the effectiveness of treatments. CoBRA [23] provides tools for quantitative risk evaluation and consulting. Using CoBRA developers reduce the losses that might result from security problems.

Risk-based requirements elicitation and prioritization (RiskREP) [10] is an iterative process for managing IT security risks. It combines the results of requirements analysis and risk analysis. The analysis is carried on in four steps: elicitation of quality goals, security risk analysis, countermeasure definition, and prioritisation.

In this work we situate our analysis at the *fine-grained* level in order to outline the capabilities of BPMN to deal with security. Our goals are to explore how we could apply BPMN to model security when considering business process modelling, and to suggest some potential BPMN extensions towards security. To ground our analysis we select the *domain model* for *information systems security risk management* (ISSRM) [7] [15]. The ISSRM approach also includes process guidelines that help identify the vulnerable assets, determine their security objectives, assess the risks, and elicit security requirements to mitigate these risks.

2.2 ISSRM Domain Model

Since the ISSRM domain model [7] [15] (shown in Fig. 1) is an important artefact to analyse BPMN in this paper, we will briefly introduce its major concepts.

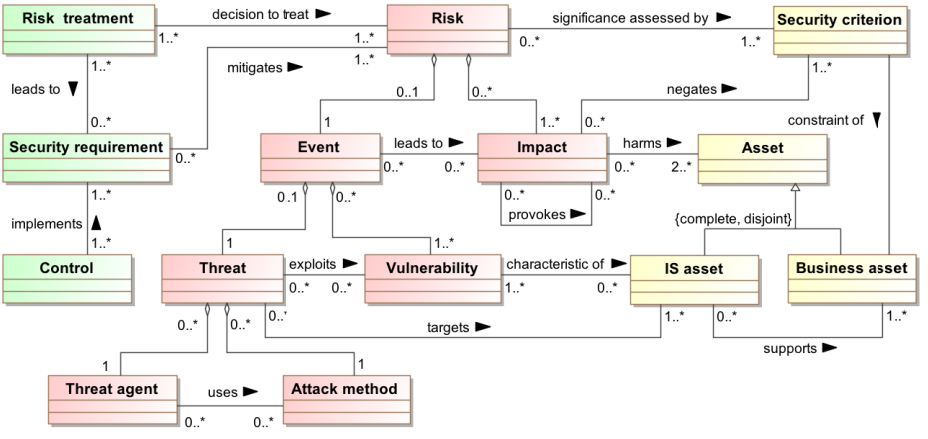

Fig. 1. The ISSRM Domain Model (adapted from [7] [15])

Assets-related concepts describe organisation's assets and their security criteria. Here, an *asset* is anything that is valuable and plays a vital role to accomplish organisation's objectives. A *business asset* describes the information, processes, capabilities and skills essential to the business and its core mission. An *IS asset* is the IS component, valuable to the organisation since it supports business assets. A *security criterion* is the property or constraint on business assets describing their security needs, which are, typically, expressed through *confidentiality*, *integrity* and *availability.*

Risk-related concepts introduce a risk definition. A *risk* is composed of a threat with one or more vulnerabilities that leads to a negative impact on one or more assets by harming them. An *impact* is the consequences of an event that negates the security criterion defined for business assets in order to harm assets. An *event* is an aggregation of threat and one or more vulnerabilities. A *vulnerability* is the characteristics of IS

assets that expose weakness or flaw. A *threat* is an incident initiated by a threat agent using attack method to target one or more IS assets by exploiting their vulnerabilities. A *threat agent* is an agent who has means to harm intentionally IS assets. An *attack method* is a standard means by which a threat agent executes threat.

Risk-treatment related concepts describe the concepts to treat risk. A *risk treatment* is a decision (e.g., *avoidance*, *reduction*, *retention*, or *transfer*) to treat the identified risk. A *security requirement* is the refinement of a risk treatment decision to mitigate the risks. A *control* designates a means to improve the security by implementing the security requirements.

Application guidelines. The ISSRM application follows the general risk management process. It is based on the existing security standards, like [3] [6] [11] [22]. It is an iterative process consisting six steps. Firstly, a developer needs to *define the organisational context and assets* that needs to be secured. Then, one *determines security objectives* (e.g., confidentiality, integrity, and availability) based on the level of protection required for the identified assets. Next, *risk analysis and assessment* help identify potential risks and their impacts. Once risk assessment is performed *risk treatment decision* should be taken. This would result in *security requirements definition*. Security requirements are *implemented* into *security controls*. The risk management process is iterative, because new security controls might open the possibility for new (not yet determined) security risks.

2.3 Research Method

The ISSRM domain model [7] [15] was developed during the *step 1* and *step 2* as illustrated in the research method in Fig. 2. The main goal of the step 1 was to identify the most important concepts of the security risk domain. The literature on the risk management standards [3] [11], security-related standards [6] [22], security risk management methods [1] [4] and software engineering frameworks [8] [9] was considered. Based on this analysis, a conceptual model (see Fig. 1) is defined. In addition each concept (i.e., class and association) is complemented with definition. In this work we focus on the third step. As discussed in [7] [15], most of the modelling languages appear to overlook security risk management (despite of few reports in [5] [13] [14]). In this paper we report on the BPMN means to address security risk management. The outcome of our analysis is the direct input for the fourth step where the modelling language could be extended with the security risk management constructs and its usage adjusted to the guidelines of the risk management process. This work is a part of the larger effort to develop a systematic model transformation-based security risk-driven method for secure system development.

2.4 BPMN

The application of BPMN modelling is divided into three model usage levels [21]. *Analytical modelling* describes the activity flow. *Executable modelling* is targeted to the system developing. In this paper our scope is *descriptive modelling*, which concentrates on business process by documenting the major business flows.

The major BPMN graphical constructs (concrete syntax) for the *descriptive modelling* are listed in Fig. 3. The extract of the BPMN abstract syntax (based on [18]) classifies BPMN graphical constructs and illustrate the relationships between them in Fig. 4 and 5.

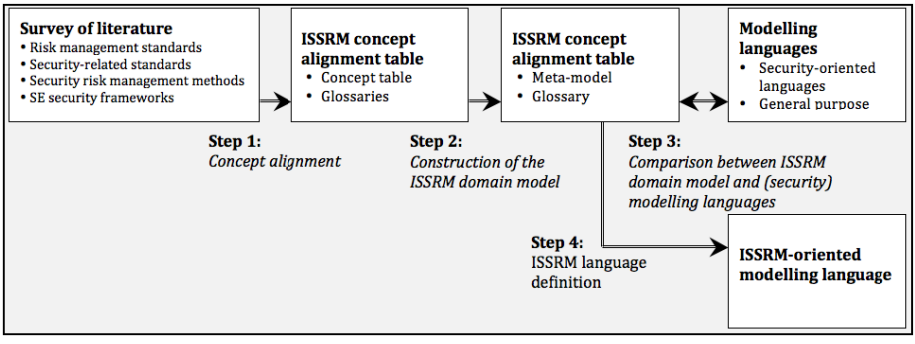

Fig. 2. A Research Method for ISSRM-oriented Modelling Languages (adapted from [15])

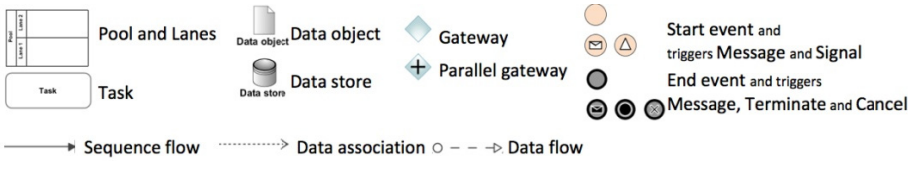

Fig. 3. BPMN Concrete Syntax (Descriptive Modelling)

BPMN includes four major categories of constructs (Fig. 4): *flow objects*, *containers*, *flows* and *artefacts*. The *flow objects* describe the atomic units of a process using *events*, *tasks* and *gateways*. An *event* indicates *start* or *end* of a process path; it can be *triggered* or *non-triggered*. A *task* is an atomic activity that has no internal sub-parts defined by the model. In some cases, the *task* can also represent the sub-process, a compound activity with sub-parts. The control of the divergence and convergence of sequence flows is realised by the *gateways*. The BPMN *containers* are *pools* and *lanes*. They both play a role of object holders. However, the *pool* shows the message flow between the process and external participants. The *lane* is a subdivision of a process used to organise flow elements belonging to different categories, and also represents a performer role or an organisational unit. The BPMN *artefacts* include such constructs as *data objects*, *data stores* and *annotations*. *Data objects* define what data is required or produced by activities. *Data stores* describe how data are stored.

Relationships (Fig. 5) between different BPMN constructs are defined using *flows*, which include *sequence flows*, *data flows*, and *data association flows*. For instance, the *sequence flows* link together the BPMN activities, gateways, and events within a single pool. The *data flows* show the input/output between pools. Finally, the *data association flows* link together the BPMN tasks and artefacts (i.e., data objects, data stores, and annotations).

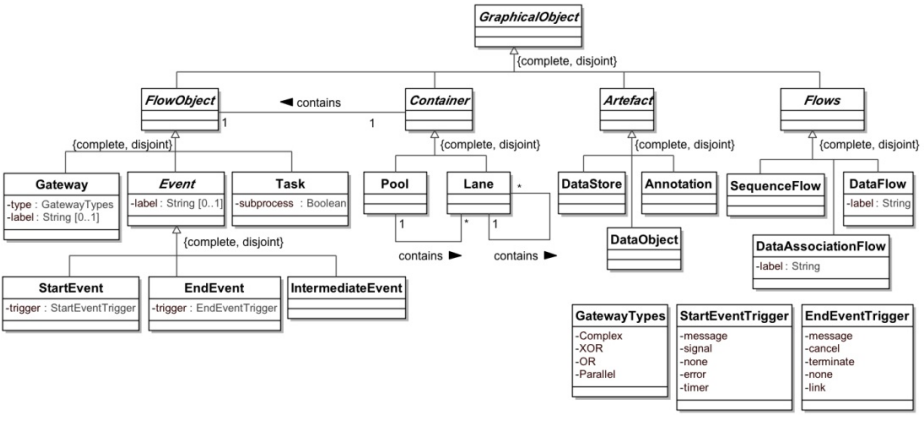

Fig. 4. The BPMN Abstract Syntax: Concept Classification

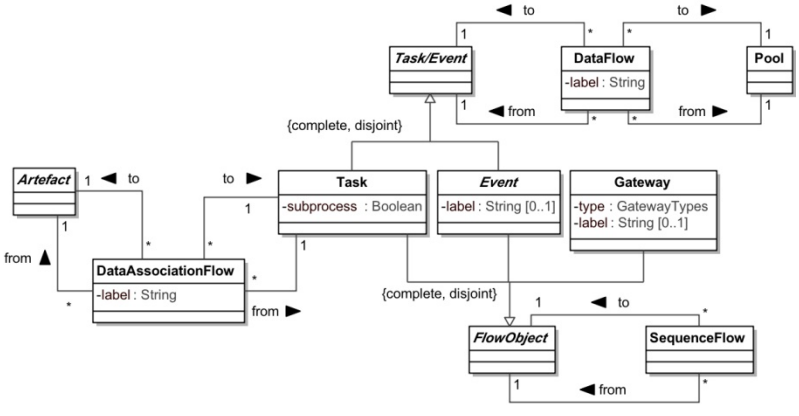

Fig. 5. The BPMN Abstract Syntax: Relationships[1]

3 Security Risk Modelling with BPMN

In this section we will follow the ISSRM process to investigate security risks in a running example modelled using BPMN. We will show which BPMN constructs could be used to address concepts of the ISSRM domain model. Our running example is an *online registration process of the Internet store*.

Context and Asset Identification. Let's consider the following situation where the potential User (*pool* User in Fig. 6) wishes to start using the Internet store system (*pool* System). In order to get registration details, user sends a message with an inquiry to the

[1] Here we do not define the explicit integrity constraints of the abstract syntax. But these exist, especially, to strengthen the flow relationships. For instance, the data association flow could only be defines between the artefacts and task; the data flow could only be defined between the pool and task/event, and similar.

system administrator. After the message is accepted (*task* Accept message) and read (*task* Read message) by the administrator, the guidelines (*data flow* Demand for registration) are sent (*task* Send answer) back to the user.

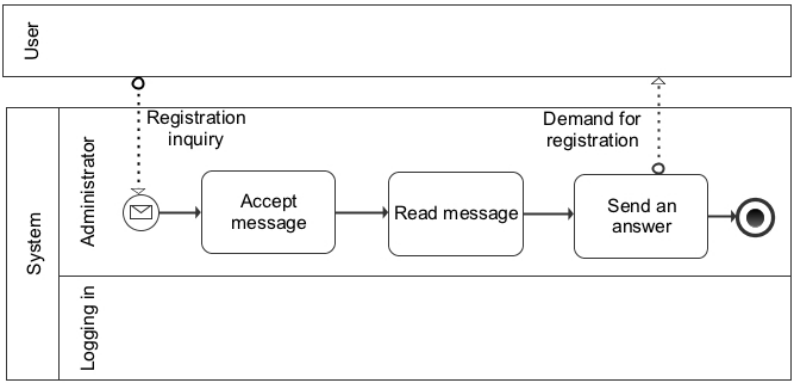

Fig. 6. Message Handling Process

In Fig. 7 we present a user registration process. After receiving the guidelines, the user registers to the Internet store by submitting his data (*data flow* User info). The system, then, accepts registration information (which includes data on the preferred Username and Password) and includes it into the database (*task* Insert data to DB).

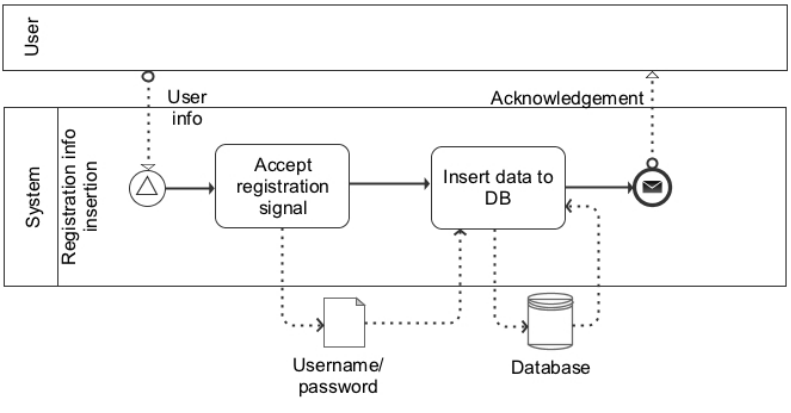

Fig. 7. User Registration Process

After registering the valid Username and Password, the user is able to login to the system. The system checks the username and the password. If these match, the user gets the *success* signal and is able to use the Internet store system. Otherwise the user gets a notification about the failure.

Determination of Security Objectives. In this scenario we can identify several major assets that needs protection against security risks. Firstly, we need to ensure *confidentiality of username and password*. If confidentiality is revealed the system

violators could use the user's personal data for not intended purposes. In addition we need to ensure *integrity of all the business processes*. If integrity is broken the system might be used not according to its purpose.

Risk Analysis and Assessment. In Fig. 8 we model a potential security risk scenario. Let's say, that there exists a violator (presented as the BPMN *pool* Violator) who would like to login to the system without registering his personal user account (skipping process defined in Fig. 7). Similarly as illustrated in Fig. 6, the violator sends a message to the system. But this time the message includes a spy program (*data flow* Message containing a spy program), which is started after the message is accepted (*task* Accept message) and read (*task* Read message). The spy program initialises a new task (e.g., Extract data from database), which sends an inquiry to the database and extracts the Usernames and Passwords of existing users. These data are then attached to a reply message, which is sent to the violator (*task* Sends an answer and *data flow* Demand for registration + data copied from database).

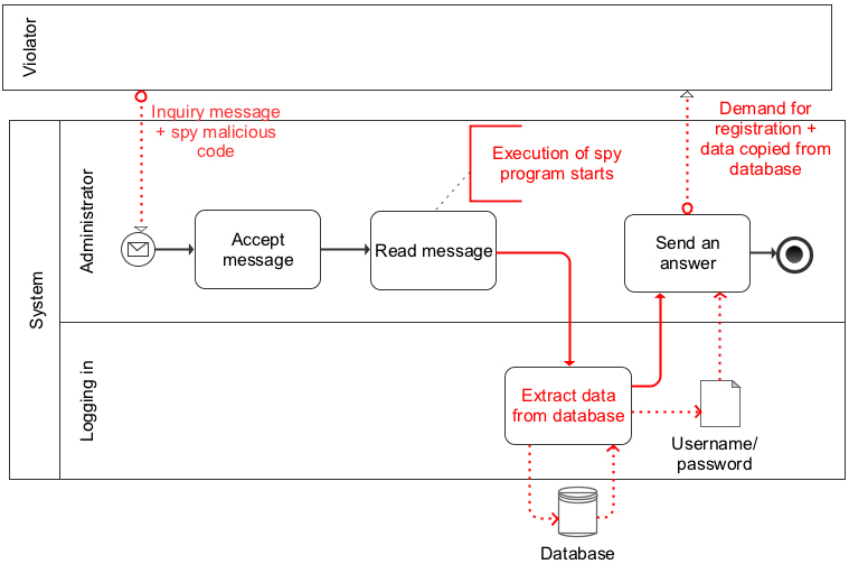

Fig. 8. Message Handling Process Including Security Risk Attack

In this analysis we are able to identify the ISSRM *threat agent* (e.g., Violator) and the ISSRM *attack method* (e.g., Message containing a spy program and Extraction of data from the database). Combination of these elements forms a security *threat*. The direct impact of this threat is that the *confidentiality* of the Usernames and Passwords is broken. On the other hand, this ISSRM *impact* provokes another *impact*, which negates the *integrity of the business processes*; i.e., the Violator is able now to access the system without registering, and, thus, change the business processes according to his needs.

Risk treatment involves deciding how the identified security flows could be mitigated. In our example we will take a *risk reduction* – i.e., actions to lessen the probability of the negative consequences – decision.

Security requirements definition. To reduce the probability of accepting the message, which contains a spy program, firstly, we introduce a *task* for Message scanning, as defined in Fig. 9. If scanning of the message reports a problem, the message is deleted and the message sender is blocked (*task* Block user/Delete message). Secondly, another security requirement includes the *task* Control activity of DB access. If there is a try to access the Database during the message handling process, it is blocked (*task* Block DB access). The final security requirement includes control of the outgoing/sent information (*task* Out-coming traffic control). This investigates if the response message is of the same length as initially defined. If this check reports a problem, the system stops the message sending (*cancel end event* Operation stopped).

Control implementation. The BPMN application is typically performed at the system analysis stages. Thus, implementation of the security requirements remains postponed for the later system development stages. On the other hand the iteration of the ISSRM process is needed where the current security requirements (e.g., ones introduced in Fig. 9) would be investigated for the new security risks.

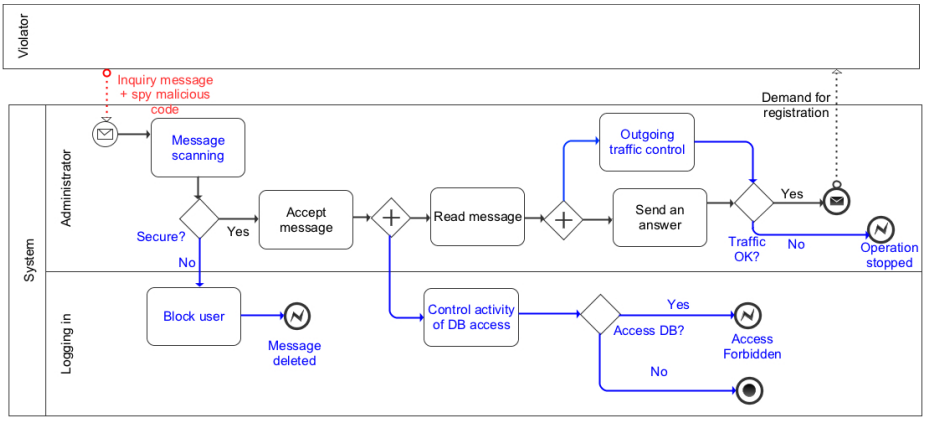

Fig. 9. Message Handling Process Including Security Requirements

4 ISSRM and BPMN Alignment

The running example illustrates a semantic alignment between ISSRM and BPMN. We show how BPMN is applied to consider possible attack scenarios and how countermeasures are defined. We summarise this discussion in Table 1.

Asset-Related Concepts. As described in Section 2, the ISSRM *business asset* could include valuable processes and information. In the first place the BPMN approach is meant for describing business processes within organisation. Thus, we can observe its constructs, such as *task, gateway*, *event* and their connecting link, i.e., *sequence flow*, that they help describing valuable processes. In the BPMN model the *flow objects* (i.e., *task, gateway* and *event*) are contained in the BPMN *containers*; i.e., *pools* and *lanes*. In other words the *container* constructs support definition and execution of the *business processes*. In terms of ISSRM, we align the *pool* and *lane* constructs to the ISSRM *information system assets*. The BPMN *data object*, which describes the

required or produced data, is aligned to the ISSRM *business asset*, and BPMN *data store* is defined as ISSRM *IS asset*.

Table 1. Alignment of the ISSRM Concepts and the BPMN Constructs

The ISSRM domain model		**BPMN constructs**	**Example**
Asset-related concepts	Asset	–	–
	Business asset	*Data object*; *Task*, *Gateway*, *Event*, *Sequence flow*	Username and Password; Processes of Message handling, User registration, and User login to the system
	IS asset	*Data store* *Pool, Lane*	Database; System, Database connection, Message
	Security criterion	–	*Confidentiality* of Usernames and Password; *Integrity* of processes for Message handling, User registration and User login to the system
Risk-related concepts	Risk	–	–
	Impact	–	Confidentiality of Usernames and Password is broken; Integrity of processes is negated
	Event	–	–
	Threat	A combination of constructs for *Threat agent* and *Attack method*	A violator sends a message containing a spy program, which extract info from database and sends it back to the violator.
	Vulnerability	–	Message is being handled without any scanning; The outgoing traffic is not monitored; The access to database is not controlled
	Threat agent	*Pool*	Violator
	Attack method	*Task*; *Flows (e.g., Data flow* with the label describing attack method; *Data association flow* with the label describing attack method);	Extract info from database; *Data flow* Message containing a spy program; *Data association flows* Sends a request and Gets data
Risk treatment related concepts	Risk treatment	–	Reduction (but other decision are also possible)
	Security requirement	*Task*, *Gateway*, *Event*, *Sequence flow*	*Tasks* Message scanning; Block user/Delete message; Control activity of DB access; Block DB access; Stop operation; Outgoing traffic control *Gateways* Secure?; Access to DB?; Traffic ok? *Events* Message deleted; Access forbidden; Operation stopped
	Control	–	–

The BPMN approach does not contain any constructs for explicit definition of the ISSRM *security criterion*. However, the created model can suggest the implicit expression (e.g., *Confidentiality of username and password*; *Integrity of the process*).

Risk-related concepts present what major principles should be taken into account when defining the potential risks. In principle the BMPN does not have the direct means to model security risks. However, in our example we have applied BPMN to model the negative and harmful processes. We have observed that the BPMN *pool*, when represents a negative/not intended actor, could be characterised as the ISSRM *threat agent*. Thus, the means that the *threat agent* is capable to use, are considered as the ISSRM *attack method*. For example, the BPMN *task*, as an atomic activity, when initialised by the "non-intended" actor, should be understood as the "means by which a threat agent executes threat"; such a *task* is aligned to the ISSRM *attack method*. Similar argumentation could be done about the BPMN *flow* and *data association flow*, which are also aligned to the ISSRM *attack method*.

We have not identified any explicit BPMN constructs to model the ISSRM *risk*, *impact*, *event*, or *vulnerability*. But we have observed that some of these concerns could be identified implicitly from the analysed problem. For instance, we can describe the ISSRM *threat* as the combination of the *threat agent* and *attack method* (see Table 1). Furthermore, two system *vulnerabilities* (namely, Message is being handled without any scanning and The outgoing traffic is not monitored) are identified. The third *vulnerability* (i.e., The access to database is not controlled) is found regarding the *database*. Finally, we can also define implicitly the ISSRM *impact*, which constitutes the negation of the identified *security criteria* and harm to the corresponding *assets*. These implicitly identified examples could not be expressed with the BPMN constructs.

Risk treatment-related concepts describe the decisions that should be taken, and controls to be implemented in order to mitigate the identified risks. In our example we select the *risk reduction*. However, other types of ISSRM *risk treatment decision* could also be taken depending on the level of risks mitigation.

The ISSRM *security requirements* are presented using the BPMN *task*, *gateway*, and *event* constructs connected using *sequence flow* links. For instance, the *security requirement* to mitigate the vulnerability Message is being handled without any scanning, starts with the BPMN *task* Message scanning, followed by the *gateway* Secure?. If the problem is found the *task* Block user/Delete message, and the process finishes with the *event* Message deleted. We do not align any BPMN construct to the ISSRM *controls*. However, we should note that in late system development stages the combination of the BPMN *task*, *gateway*, and *event* constructs (as illustrated above) might result in different security control modules.

5 Discussion and Conclusion

Our major contribution is the semantic alignment of the BPMN constructs to the ISSRM concepts. In addition we define a way to elicit security requirements for the important business processes. In this section we discuss validity, conclude the study with the potential BPMN extensions, and present the related and future work.

5.1 Threats to Validity

Our results contain a certain degree of subjectivity. Two researchers have performed this study. Thus, it might mean that some aspects of the BPMN approach or its application could be interpreted and aligned to the ISSRM concepts differently. Also, the running example involves the subjective decisions on how problem needs to be modelled. For instance, we have taken a risk reduction decision. The security requirements would be different if one would take the risk avoidance decision.

The scope of the current work is limited to the BPMN *descriptive modelling*. We acknowledge the importance to investigate the *analytical* and *executable* modelling, but this remains for the future research. Finally, in this work we analyse only a simple example of the Internet store. Although this example is realistic, we have not applied it in the practical settings. Thus, our analysis remains based on the selected BPMN literature [18] [21] [24].

5.2 BPMN Extensions towards Security Risk Management

In general, the BPMN approach is not specifically dedicated to the security modelling but to the business process modelling. On one hand we argue that the major version of the language should not loose its original purpose, and it should remain relatively simple. On the other hand we illustrate that BPMN provided the major set of constructs that help understanding important business assets, their security risks, and potential security requirements. Certainly this requires some potential language extensions:

- Using BPMN we are able to address only a part of the ISSRM domain model. For example, we were not able to express the ISSRM *security criterion*, *risk*, *impact*, *vulnerability*, *risk treatment*, and *control* constructs. This situation suggests potential extensions of the BPMN approach (at the concrete syntax, abstract syntax and semantic levels) and this is a potential direction for future research.
- The same constructs used for different ISSRM concepts. This could be noticed for the BPMN *task*, which is used to express the ISSRM *business asset*, *attack method*, and *security requirement* constructs; the BPMN *pool*, which helps modelling the ISSRM *threat agent* and *IS asset* constructs; and also some other constructs and links. This situation might provoke a readability and comprehensibility problem. There might be few solutions. The modellers could apply meta-labelling to identify different ISSRM-related concepts (e.g., [Business asset], [Attack method], or [Security requirement]) or introduce differentiating variables (e.g., *white* for the *asset-related*, *red* for the *risk-related*, and *blue* for the *treatment-related* constructs) between the same BPMN constructs aligned to different ISSRM constructs.

During our analysis we faced with a problem when one ISSRM concept could be presented using several BPMN constructs. For example, the ISSRM *security requirement* is modelled using the combination of the ISSRM *task*, *gateway*, *event* constructs and *sequence flow* links. This makes it difficult to understand the heuristics of the modelling process. Thus, it could be helpful to define rules and/or patterns to guide the use of the (security) modelling constructs.

5.3 Related Study on Security-Oriented BPMN

In [19] Rodríguez *et al.* propose the BPMN extensions for modelling secure business processes through understanding the security requirements. Firstly, their proposal illustrates the extension of the BPMN abstract syntax with the security-related concepts such as non-reputation, attack harm detection, integrity, privacy, access control, security role and security permission. Secondly, the concrete BPMN syntax is extended through the stereotypes introduced to the ordinary constructs of BPMN. The study does not include any consideration of the extension semantics. Further, in [20] some extensions of BPMN (called *BPSec*) are proposed towards the graphical representation of security requirements. They present a symbol of *padlock* to express security requirements and a *padlock with twisted corner* for audit register.

In [16] Menzel *et al.* proposes the BPMN enhancements towards trust modelling. They focus on the outline the metric that describes the value of enterprise assets and pay attention to the level of security or so called trust level of each participant of the process. Here, enterprise assets are presented using BPMN tasks, data objects, and communication links between tasks and participants. Authors define how to enable trustworthy interactions, organisational trust, and security intensions through BPMN. Other proposed extension is a security policy model used to define specific security patterns for authorisation, authentication, integrity, and confidentiality.

The limitations of these works [16] [19] [20] are that they focus either on a coarse-grained level, or target only some security aspects in business processes. In comparison our study does not propose any BPMN extensions. However, we present a semantically grounded fine-grained analysis based on the well-established ISSRM domain model [7] [15]. As a result we present the alignment between ISSRM concepts and the BPMN constructs, which allows developers to understand current BPMN means to deal with security. Also we identify potential BPMN extensions towards security both at the (concrete and abstract) syntax and at the security risk-oriented semantics levels. In other words we explore the reasons *why* and *how* BPMN needs to be extended to consider security at the business process modelling.

Paja *et al*. introduce a method to understand *security needs* through participants' objectives and interactions [17]. *Security requirements* are captured in terms of social commitments between the actors of the system. Then these security requirements are used to annotate business processes modelled in BPMN. Similarly, in our proposal we argue that security annotated BPMN models could be further analysed using the same modelling language, namely BPMN. The advantage is that the business analyst would not be required to learn yet another modelling notations, but would be able systematically reason for the return on security investment in business processes.

5.4 Related Studies on Security Risk-Oriented Modelling Languages

BPMN is not the only language assessed for the IS security risk management: ISSRM has been used to evaluated Secure Tropos [14], misuse cases [13], KAOS extensions to security [15], and Mal-activity diagrams [5]. But BPMN is the language to define the business process modelling. We have not found any business modelling language, which would support security analysis; thus the recent standard [24] for business process modelling was our natural choice. We envision that after analyzing a number of languages for security modelling it will be possible to facilitate model transformation and interoperability between them, thus introducing the security

analysis from the early development stages to design and implementation, also resulting in a sustainable and secured system. Such a model transformation would be supported by transformation rules, developed on the semantic alignment of the (*business* and *security*) modelling approaches to the common base, i.e., the ISSRM domain model. However, definition of these transformation rules also remains a future study.

References

1. Alberts, C.J., Dorofee, A.J.: OCTAVE Method Implementation Guide Version 2.0. Carnegie Mellon University - Software Engineering Institute, Pennsylvania (2001)
2. Asnar, Y., Giorgini, P., Massacci, F., Zannone, N.: From Trust to Dependability through Risk Analysis. In: Proceedings of ARES 2007, pp. 19–26. IEEE Computer Society (2007)
3. AS/NZS 4360, Risk management. SAI Global (2004)
4. Braber, F., Hogganvik, I., Lund, M.S., Stølen, K., Vraalsen, F.: Model-based Security Analysis in Seven Steps—a Guided Tour to the CORAS Method. BT Technology Journal 25(1), 101–117 (2007)
5. Chowdhury, M.J.M., Matulevičius, R., Sindre, G., Karpati, P.: Aligning Mal-activity Diagrams and Security Risk Management for Security Requirements Definitions. In: Regnell, B., Damian, D. (eds.) REFSQ 2011. LNCS, vol. 7195, pp. 132–139. Springer, Heidelberg (2012)
6. Common Criteria version 2.3, Common Criteria for Information Technology Security Evaluation, CCMB-2005-08-002 (2005), `http://www.tse.org.tr/turkish/belgelendirme/ortakkriter/ccpart2v2.3.pdf`
7. Dubois, E., Heymans, P., Mayer, N., Matulevičius, R.: A Systematic Approach to Define the Domain of Information System Security Risk Management. In: Intentional Perspectives on Information Systems Engineering, pp. 289-306. Springer (2010)
8. Firesmith, D.G.: Engineering Safety and Security Related Requirements for Software Intensive Systems. In: Companion to the Proceedings of the 29th International Conference on Software Engineering (COMPANION 2007), p. 169. IEEE Computer Society (2007)
9. Haley, C.B., Laney, R.C., Moffett, J.D., Nuseibeh, B.: Security Requirements Engineering: A Framework for Representation and Analysis. IEEE Transactions on Software Engineering 34, 133–153 (2008)
10. Herrmann, A., Morali, A., Etalle, S., Wieringa, R.: Risk and Business Goal Based Security Requirement and Countermeasure Prioritization. In: Niedrite, L., Strazdina, R., Wangler, B. (eds.) BIR Workshops 2011. LNBIP, vol. 106, pp. 64–76. Springer, Heidelberg (2012)
11. ISO/IEC Guide 73, Risk management - Vocabulary - Guidelines for use in standards. International Organization for Standardization, Geneva (2002)
12. Jürjens, J.: Secure Systems Development with UML. Springer, Heidelberg (2005)
13. Matulevičius, R., Mayer, N., Heymans, P.: Alignment of Misuse Cases with Security Risk Management. In: Proceedings of ARES 2008, pp. 1397–1404. IEEE (2008)
14. Matulevičius, R., Mayer, N., Mouratidis, H., Martinez, F.H., Heymans, P., Genon, N.: Adapting Secure Tropos for Security Risk Management in the Early Phases of Information Systems Development. In: Bellahsène, Z., Léonard, M. (eds.) CAiSE 2008. LNCS, vol. 5074, pp. 541–555. Springer, Heidelberg (2008)
15. Mayer, N.: Model-based Management of Information System Security Risk. Doctoral Thesis, University of Namur (2009)

16. Menzel, M., Thomas, I., Meinel, C.: Security Requirements Specification in Service-oriented Business Process Management. In: ARES 2009, pp. 41–49 (2009)
17. Paja, E., Giorgini, P., Paul, S., Meland, P.H.: Security Requirements Engineering for Secure Business Processes. In: Niedrite, L., Strazdina, R., Wangler, B. (eds.) BIR Workshops 2011. LNBIP, vol. 106, pp. 77–89. Springer, Heidelberg (2012)
18. Remco, M., Dijkman, R.M., Dumas, M., Ouyang, C.: Formal Semantics and Analysis of BPMN Process Models using Petri Nets. Queensland University of Technology, TR (2007)
19. Rodríguez, A., Fernández-Medina, E., Piattini, M.: A BPMN Extension for the Modeling of Security Requirements in Business Processes. IEICE – Transactions on Information and Systems E90-D(4), 745–752 (2007)
20. Rodríguez, A., Fernández-Medina, E., Piattini, M.: UbiComp 2007. LNCS, vol. 4717, pp. 408–415 (2007)
21. Silver, B.: BPMN Method and Style: A Levels-based Methodology for BPMN Process Modeling and Improvement using BPMN 2.0. Cody-Cassidy Press (2009)
22. Stoneburner, G., Goguen, A., Feringa, A.: NIST Special Publication 800-30: Risk Management Guide for Information Technology Systems. National Institute of Standards and Technology, Gaithersburg (2002)
23. Trendowicz, A.: Tutorial: CoBRA - Cost Estimation, Benchmarking and Risk Analysis Method (2005), http://www.dasma.org/metrikon2005/tutorial_cobra.pdf
24. White, S.A.: Introduction to BPMN, IBM (2004), http://www.bpmn.org/Documents/Introduction_to_BPMN.pdf