**An Extension of Business Process Model and Notation for Security Risk Management**

Olga Altuhhova, Raimundas Matulevičius* and Naved Ahmed

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

**Abstract.** Business process modelling is one of the major aspects in the modern information system development. Recently business process model and notation (BPMN) has become a standard technique to support this activity. Typically the BPMN notations are used to understand enterprise’s business processes. However, there is a limited work to address how security concerns are addressed during the management of the business processes. This is a problem, since both business processes and security should be understood in parallel to support a development of the secure information systems. In the previous work we have analysed BPMN with respect to the domain model of the IS security risk management (ISSRM) and showed how the language constructs could be aligned to the concepts of the ISSRM domain model. In this paper we propose the BPMN extensions for security risk management based on the BPMN alignment to the ISSRM concepts. We illustrate how the extended BPMN could express assets, risks and risk treatment on few running examples related to the Internet store regarding the asset confidentiality, integrity and availability. 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.

**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 competitiveness 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, Jurjens (2005) observes 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 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 (Menzel et al., 2009; Rodriguez et al., 2007a, 2007b), information assurance and security (Cherdantseva et al., 2012) or to relate business process and security requirements modelling (Paja et al., 2012), 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) (Remco et al., 2007; Silver, 2009), a multi-vendor standard controlled by the Object Management Group (White, 2004). 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. This paper is an extension of our previous report (Altuhhova et al., 2012), where we have selected a domain model (Mayer, 2009; Dubois et al., 2010) for IS Security Risk Management (ISSRM) and aligned the BPMN constructs to the concepts of this domain model. We have resulted in a grounded and fine-grained reasoning for extensions of BPMN toward secure business processes. Based on this alignment, in this paper we introduce a set of security risk-oriented extensions for BPMN. In this paper our goal is to illustrate (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. We result in the security risk-aware BPMN, which could be used to express secure business assets, potential security risks, and their countermeasures. We illustrate our analysis and proposal through few running examples regarding the asset confidentiality, integrity and availability. In this way we end up with guidelines for the BPMN application to analyse security risks.

The paper structure is as follows: firstly, we give the background to our study. Next, we present concrete and abstract syntax of the proposed BPMN security extensions, and illustrate them through confidentiality, integrity and availability analysis in the Internet store example. Then we discuss threats to validity, overview the related work, and conclude the study.

BACKGROUND

Security Analysis Methods

To model secure systems, different security risk management approaches are developed. For instance, CORAS is a model-driven approach (Braber et al., 2007), which includes a systematic guidance for security risk analysis. The Tropos Goal-Risk framework (Asnar et al., 2007) 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 (Trendowicz, 2005) 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) (Herrmann et al., 2012) is an iterative process for managing IT security risks. It combines the results of requirements analysis and risk analysis. The process 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 managing business processes, and to suggest some potential BPMN extensions towards security. In our work we have selected the ISSRM domain model (Mayer, 2009; Dubois et al.,
It suggests the process guidelines that help identify the vulnerable assets, determine their security objectives, assess the risks, and elicit security requirements to mitigate these risks.

**ISSRM Domain Model**

Since the ISSRM domain model (Mayer, 2009; Dubois et al., 2010) (shown in Figure 1) is an important artefact to analyse BPMN, we will briefly introduce its major concepts.

**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* 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 (AS/NZS 4360, 2004; Common Criteria, 2005; ISO/IEC Guide 73, 2002; Stoneburner et al., 2002). 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.

Research Method

The ISSRM domain model (Mayer, 2009; Dubois et al., 2010) was developed during the step 1 and step 2 as illustrated in the research method in Figure 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 (AS/NZS 4360, 2004; ISO/IEC Guide 73, 2002), security-related standards (Common Criteria, 2005; Stoneburner et al., 2002), security risk management methods (Alberts and Dorofee, 2001; Braber et al., 2007) and software engineering frameworks (Firesmith, 2007; Haley et al., 2008) was considered. Based on this analysis, a conceptual model (see Figure 1) is defined. In addition each concept (i.e., class and association) is complemented with definition.

In (Altuhhova et al., 2012) we have reported on the BPMN means to address security risk management (step 3). We observed that BPMN overlooks security risk management since the language is not specifically designed for the security modelling. Our study resulted in a number of language limitations, summarised in the next section. The outcome of the analysis (Altuhhova et al., 2012) is the input for step 4 where the BPMN could be extended with the security risk management constructs and its usage adjusted to the guidelines of the risk management process. This is our primary goal in this paper. This work is a part of the larger effort to develop a systematic model transformation-based security risk-driven method for secure system development.

Figure 2. A Research Method for ISSRM-oriented Modelling Languages

Business Process Model and Notation

The application of BPMN modelling is divided into three levels based on the usage (Silver, 2009). 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 graphical constructs (concrete syntax) for the descriptive modelling are listed in Figure 3. In this section we will summarise the previously performed alignment (Altuhhova et al., 2012) of the BPMN constructs to the concepts of the ISSRM domain model.
Asset-related concepts. In the first place the BPMN approach is meant for describing business processes within organisation. Altuhhova et al. (2012) observe that its constructs, such as task, gateway, event and their connecting link, i.e., sequence flow, help describing valuable processes (i.e., ISSRM business assets). The flow objects (such as 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, the pool and lane constructs are aligned 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 the ISSRM IS asset.

Risk-related concepts. BPMN does not contain the direct means to model security risks. However, in (Altuhhova et al., 2012) BPMN is applied to model the negative and harmful processes. Then the BPMN pool, when it represents a negative/not intended actor, could be characterised as the ISSRM threat agent. Thus, the means (e.g., BPMN tasks, flow and data association flow) that the threat agent is capable to use, are considered as the ISSRM attack method. There were no explicit BPMN constructs to model the ISSRM risk, impact, event, or vulnerability. But some of these concerns could be understood implicitly from the analysed problem. For instance, it is possible to define the ISSRM threat as the combination of the threat agent and attack method.

Risk treatment-related concepts. The ISSRM security requirements are presented using the BPMN task, gateway, and event constructs connected using sequence flow links. However there is no BPMN construct to express the ISSRM controls. It was also noted that in late system development stages the combination of the BPMN task, gateway, and event constructs might result in different security control modules (however not modelled using BPMN at the descriptive modelling).

To conclude, the Altuhhova’s et al. (2012) overview, BPMN approach is not specifically dedicated to the security modelling but to the business process modelling. On one hand, the major version of the language should not lose its original purpose, and it should remain relatively simple. On the other hand BPMN provides the major set of constructs that help understanding important business assets, their security risks, and potential security requirements. Certainly, this requires some language extensions as illustrated in the next section.
SECURITY RISK-AWARE BPMN

Concrete Syntax

Moody (2009) argues for the importance of visual variables (e.g., shape, size, colour, etc) when defining visual syntax of the modelling languages. One of our proposals to improve the BPMN language for the security risk analysis is the introduction of different colours for the three groups of constructs: (i) black for the asset related constructs; (ii) red for the risk-related constructs; and (iii) blue for the risk treatment-related constructs.

Table 1 lists our proposal for the asset related constructs. Altuhhova et al (2012) indicated the importance to differentiate between meaning of the BPMN constructs when expressing different ISSRM concepts. For example, the BPMN task could be used to express both the ISSRM business assets and IS assets. In order to separate this we introduce two icons as illustrated in Table 1. Additionally we also present a visual element – lock – to express the ISSRM security objective. The lock is placed (as the constraint of) on the business asset, representing its security needs. The security criterion is defined, then, in the annotation associated to the lock construct.

Concepts to express the risk-related constructs are presented in Table 2. The ISSRM threat agent could be expressed using the BPMN containers, i.e., pools and lanes, and the ISSRM attack method is defined as the combination of flow objects (i.e., event, gateway, and task) using sequence flows.

Table 2 shows that vulnerability could be defined using annotations, which are assigned to the vulnerability point. This point is defined as the characteristic of the IS asset (see Table 1). Further, we also introduce the notion of the ISSRM impact through the unlock symbol. If the security criterion is negated then the security objective (defined using lock as presented in Table 1) is broken. The appropriate BPMN relationships (see Table 2, leads to relationships) are used to define how risk harms the business asset(s) and IS asset(s).

Following the domain model, the ISSRM threat is defined as a combination of the BPMN constructs used to model threat agent and attach method; the ISSRM event is expressed through the combination of constructs for threat and vulnerability. The ISSRM risk is modelled using the BPMN constructs for event and impact.

Table 3 presents the BPMN constructs used to express the ISSRM risk treatment-related constructs. In principle it introduces the combination of flow objects (i.e., event, gateway, and task) used to model the ISSRM security requirements and mitigation relationship. Other ISSRM constructs are not explicitly expressed because (i) the risk treatment is rather a mental decision done towards the mitigation of the identified risk, and (ii) security control is a part of the system implementation stage (but not analysis, where BPMN is typically applied).
Table 1. Concrete Syntax of the Security Risk-aware BPMN: Asset-related Concepts

<table>
<thead>
<tr>
<th>ISSRM (R – relationship; C – concept)</th>
<th>BPMN constructs</th>
<th>Concrete syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Asset</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>Combination of Flow Objects (Event, Gateway, Tasks) using sequence flow.</td>
<td><img src="image" alt="Concrete Syntax Diagram" /></td>
</tr>
<tr>
<td>For Business assets</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
<td></td>
</tr>
<tr>
<td>For IS assets</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IS</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Business asset</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>Data object</td>
<td><img src="image" alt="Concrete Syntax Diagram" /></td>
</tr>
<tr>
<td><strong>IS asset</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>Data store</td>
<td><img src="image" alt="Concrete Syntax Diagram" /></td>
</tr>
<tr>
<td>Containers (Pool and Lanes)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Supports</strong></td>
<td>Implicitly:</td>
<td><img src="image" alt="Concrete Syntax Diagram" /></td>
</tr>
<tr>
<td>R</td>
<td>Container (IS asset) supports combination of Flow Objects (Business assets) by containing them.</td>
<td></td>
</tr>
<tr>
<td>R</td>
<td>Sequence flow between Flow Objects (IS assets) and Flow Objects (Business assets)</td>
<td><img src="image" alt="Concrete Syntax Diagram" /></td>
</tr>
<tr>
<td>R</td>
<td>Data Association Flow between Task (IS asset) and Data Object (Business asset) and between Data Store (IS asset) and Task (Business asset)</td>
<td><img src="image" alt="Concrete Syntax Diagram" /></td>
</tr>
<tr>
<td><strong>Constraint of</strong></td>
<td>Lock and Association Flow that points from the Lock to an Annotation. Lock is a property of constructs that describe Business assets (Data Objects and Tasks)</td>
<td><img src="image" alt="Concrete Syntax Diagram" /></td>
</tr>
<tr>
<td><strong>Security objective</strong></td>
<td>Is a property of a Lock that can have a value:</td>
<td><img src="image" alt="Concrete Syntax Diagram" /></td>
</tr>
<tr>
<td>C</td>
<td>c – confidentiality</td>
<td></td>
</tr>
<tr>
<td>i – integrity</td>
<td></td>
<td></td>
</tr>
<tr>
<td>a – availability</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Security criterion</strong></td>
<td>Annotation</td>
<td><img src="image" alt="Concrete Syntax Diagram" /></td>
</tr>
<tr>
<td>ISSRM (R – relationship; C – concept)</td>
<td>BPMN constructs</td>
<td>Concrete syntax</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>-----------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>Risk</td>
<td>C Combination of Event and Impact</td>
<td><img src="image" alt="Diagram" /></td>
</tr>
<tr>
<td>Significance assessed by</td>
<td>R -</td>
<td>-</td>
</tr>
<tr>
<td>Event</td>
<td>C Combination of constructs for Threat and Vulnerability</td>
<td><img src="image" alt="Diagram" /></td>
</tr>
<tr>
<td>Targets / leads to (leads to a harm of IS assets)</td>
<td>R Sequence Flow from Flow Objects (Attack method) to Flow Objects (IS assets). Data Association Flow from Task (Attack method) to Data Store (IS asset). Sequence Flow and Data Association Flow both correspond to Targets and Leads to (in this case it leads to the harm of the IS assets).</td>
<td><img src="image" alt="Diagram" /></td>
</tr>
<tr>
<td>Leads to (leads to a harm of Business assets)</td>
<td>R Sequence Flow from Flow Objects (Attack method) to Flow Objects (Business assets). Data Association Flow from Task (Attack method) to Data Object (Business asset). Leads to a potential harm of the Business asset.</td>
<td><img src="image" alt="Diagram" /></td>
</tr>
<tr>
<td>Impact/ negates/ harms</td>
<td>R/C Unlock Unlock is a property of constructs that describe the Business assets</td>
<td><img src="image" alt="Diagram" /></td>
</tr>
<tr>
<td>Threat</td>
<td>C Combination of construct for Threat Agent and Attack method</td>
<td><img src="image" alt="Diagram" /></td>
</tr>
<tr>
<td>Exploits</td>
<td>- -</td>
<td>-</td>
</tr>
<tr>
<td>Vulnerability</td>
<td>C Annotation</td>
<td><img src="image" alt="Diagram" /></td>
</tr>
<tr>
<td>Characteristics of</td>
<td>R Vulnerability point and Association Flow that points to Annotation. Vulnerability point is a property of constructs that describe IS assets, i.e. Data Object and Task</td>
<td><img src="image" alt="Diagram" /></td>
</tr>
<tr>
<td>Threat agent</td>
<td>C Pool and Lane (Containers)</td>
<td><img src="image" alt="Diagram" /></td>
</tr>
<tr>
<td>Attack method</td>
<td>C Combination of Flow Objects (Event, Gateway, Task) using Sequence Flow and Data Flows</td>
<td><img src="image" alt="Diagram" /></td>
</tr>
<tr>
<td>Uses</td>
<td>R Data Flow</td>
<td><img src="image" alt="Diagram" /></td>
</tr>
</tbody>
</table>
### Table 3. Concrete Syntax of the Security Risk-aware BPMN: Risk Treatment-related Concepts

<table>
<thead>
<tr>
<th>ISSRM (R – relationship; C – concept)</th>
<th>BPMN constructs</th>
<th>Concrete syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>Risk treatment C</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Decision to threat R</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Leads to R</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security requirements and Mitigates C R</td>
<td>Combination of Flow Objects using Sequence Flow</td>
<td></td>
</tr>
<tr>
<td>Implements R</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Controls C</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Abstract Syntax

BPMN uses four major classes of constructs at the level of descriptive modelling: these are **FlowObjects**, **Containers**, **Flows** and **Artefacts** (see Figure 4). FlowObjects represent atomic units of a process, which can consist of 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.

**Figure 4. Abstract Syntax of the Security Risk-aware BPMN: Concept Classification**

Containers could be used for different object holders. These include a pool and a lane. The pool represents a participant of the process as independent units, showing the message flows between them. The pool can contain some number of lanes, each representing different parts of a working system, e.g., a performer role on the organizational unit.

The artefacts are represented by data store, data object and annotation. Data objects describe resources that travel within the process flow; i.e., data can be produced by one activity and, then, used as an input by another. To demonstrate how the data can be stored, the data store is used. Annotations are applied to give any additional textual information to the process or its components.
We propose two additional abstract concepts at the abstract syntax level. In Figure 4 the Vulnerability point is introduced as a property of a task or a data store and indicates the place of a system weakness. The lock concept is defined to express the constraint of valuable business with respect to security objective (e.g., integrity, confidentiality and availability). Lock has a value attribute, which indicates whether the security criteria is maintained (see security objective in Table 1) or negated (see impact in Table 2).

Relationships (Figure 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. The data association flows link together the BPMN tasks and artefacts (i.e., data objects, data stores, and annotations).

Figure 5. Abstract Syntax of the Security Risk-aware BPMN: Relationships (1)

As illustrated in Figure 6 the vulnerability point and lock are associated to annotations. This means, for example, that the vulnerability point indicated the place where the weakness of the system (i.e., IS asset) potentially exists and then, using annotations, one is able to define the actual vulnerability.

Figure 6. Abstract Syntax of the Security Risk-aware BPMN: Relationships (2)
of the IS asset (see, visual presentation in Table 1). Similarly, the concrete security criterion is defined in annotations and associated to the security objective expressed using security objective, i.e., locks (see Table 2).

**Application of the Security Risk-aware BPMN**

To illustrate the BPMN security extensions, in this section we will discuss three examples related to the security criteria of the business assets. The goal of these examples is not to give the complete analysis of the problem, but rather to illustrate how the introduced BPMN extensions could be applied. Our examples are related to an online registration process to the Internet store. They will discuss confidentiality, integrity, and availability of few business assets.

**Confidentiality analysis**

**Context and asset identification.** Let’s consider the following situation where the potential User (pool User in Figure 7) wishes to start using the Internet store system (pool Internet Store). In order to get registration details, the user requests for login and password (i.e., sends a message with an inquiry). After the message is registered (task Register received message) and managed (see tasks Accept message, Read message, and Prepare answer) by the administrator, the guidelines (data flow Request to register) are sent (task Send out answer) back to the user.

![Figure 7. Confidentiality Analysis - Handle Request Message](image)

In Figure 8 we present a user registration process. After receiving the request to register, the user provides his information (see data flow User info) to the Internet store system. The system, then, accepts registration information (which has data on the preferred Login and Password) and inserts it into the database (task Insert data to Database).
**Determination of security objectives.** Firstly, in this scenario we identify confidentiality of login and password (in Figure 8 see annotation, associated to the lock on the data object Login and password). If confidentiality is negated the system violators could use the user’s personal data for not intended purposes.

**Risk analysis.** In Figure 9 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 Figure 8). Similarly as illustrated in Figure 7, the violator sends a message to the system. But this time the message contains a spy program (data message flow Request for login and password and hidden spy program). The spy programs starts (see task Start spy program) after the message is accepted (task Accept message). The spy program initialises a new task (e.g., Extract login and password), which extracts logins and passwords of existing users from the database and adds them to the reply message, which is sent to the violator (see data flow Request to register + logins and passwords copied form database).

In this analysis we are able to identify the ISSRM threat agent (e.g., Violator) and the ISSRM attack method (e.g., Request for login an password and hidden spy program, Start spy program and Extract login and password). Combination of these elements forms a security threat. It is also possible to identify vulnerabilities, such as Message is handled without scanning, Access to DB is not controlled, and Outgoing traffic is not monitored. The direct impact of this threat is that the confidentiality of the Usernames and Passwords is broken (as indicated by the unlock icon).

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

**Security requirements definition.** To reduce the probability of accepting the message, which contains a spy program, firstly, we introduce a task Scan incoming message, as defined in Figure 10. 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 triggered end event Operation stopped).

Figure 9. Confidentiality Analysis - Security Risk

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 Figure 10) would be investigated for the new security risks.

We will not discuss all the details of integrity and availability analysis following the steps of the ISSRM process. However in the next sections we will briefly illustrate how these security criteria could be potentially captured using the extended BPMN notations for security risk management.
Integrity analysis

The risk identified in the confidentiality example provokes the negation of the integrity of the Internet store usage process. In Figure 11 the sub-process Using Internet store is considered as a business asset supported by the Internet store itself. It has an integrity security criterion. As illustrated in Figure 12, this subtask consists of other sub-processes, such as Handle request message (see Figure 7 for the subtask details), Register to Internet store (see Figure 8 for the subtask details), and Login to Internet store.

Figure 11. Integrity analysis – sub-process Start using Internet store

If the confidentiality of the username and password is negated (as illustrated in the above example), it will provoke the negation of the integrity of the Start using Internet store sub-process. Figure 13 shows that after acquiring the confidential data (i.e., usernames and passwords), the violator will be able to skip the sub-process Register to Internet store. As introduced in Figure 12, security requirements expressed using tasks Scan incoming message, Control activity of database, and Control outgoing traffic, could potentially mitigate the identified security risk.
Availability analysis

Regarding the availability, let’s consider Figure 14, where User is logged in the Internet store and request some item list (see, task Request item list) in order to select goods from it (see, task Select goods from the item list). This task could be done if the item list is available for the user. These tasks (i.e., business assets) are supported by two IS asset tasks Accept request for item list and Provide (display) item list.
In Figure 15 the Violator also requests for the item list. However in this example he exploits the IS system vulnerability (i.e., Number of requests is not limited) and sends the unlimited number of requests thus making the Internet store not capable to handle them. This negated the availability of the item list. This risk is an illustration of the denial of service, DoS attack (Loukas and Oke, 2010). When the number of requests exceeds that a server is not able handle, thus, negating the availability of the service.

Figure 15. Availability analysis – denial of service security risk
The mitigation of the problem is illustrated in Figure 16. The task Check for abnormal request (i.e., the security requirement) is introduced to mitigate the identified security risk (Loukas and Oke, 2010).

Figure 16. Availability analysis – security requirements

Validation

Theoretical validation. We have evaluated the extended BPMN notations according to the principle of the semiotic clarity (Wand and Weber, 1993; Opdahl and Henderson-Sellers, 2005). According to this principle, there should be a one-to-one correspondence between a visual language construct and its referent concept in the semantic domain. For example the ISSRM threat agent is expressed using the BPMN container constructs. This is, however the closest language construct and ISSRM concept (one-to-relatively one) correspondence. This means that there exist redundancy, overload, incompleteness and under-definition (excess) limitations of BPMN.

Redundancy means that two language constructs have the same or overlapping semantics. Redundancy limitations exist regarding the ISSRM assets, since different BPMN constructs (e.g., combination of flow objects and data object to model business assets; combination of flow objects, containers and data store to model IS assets) could be used to express them.

Overload exists if the same language construct has several meanings. This situation also observed regarding the ISSRM asset modelling. Here the combination of flow objects could be used to express both the business assets and the IS assets.

Incompleteness appears when a language does not convey information on a certain phenomenon. The extended BPMN for the security risk management contains few incompleteness limitations. For example, the problem exists regarding the ISSRM security criterion and vulnerability. In principle
our extensions highlights the “points” where these concepts are observed but not the actual concept expressions (traditionally comments are not considered to be parts of the modelling language). Other ISSRM constructs, e.g., risk, impact, event and threat) are not expressed using single BPMN constructs, but as combinations of the aggregating constructs. Finally, the ISSRM concepts of risk treatment and control are not modelling using BPMN, since these are outside the scope of the BPMN application.

**Under-definition** (or excess) arises when a language construct has no semantics. Although the ISSRM concepts of assets, attack method and security requirement are expressed using the combination of the flow objects, individually the BPMN task, event, gateway, or sequence flow has how semantics with respect to the ISSRM domain.

These language limitations do not necessarily mean the weaknesses of the language itself. It should be noticed that we are proposing our extension of the existing modelling language, which originally is not assumed to deal with the security risk management. We do not intend to suggest any new “security risk and business management language” that would completely correspond to the ISSRM domain. Our goal is rather to understand what must be taken into account when managing security risks using BPMN.

**Empirical validation.** Our proposal – the security risk-aware BPMN – was briefly validated in few examples. Firstly, we have applied it in several student exercises to develop the discussed examples for the security criteria analysis. Secondly, we have used these BPMN extensions to develop a set of security risk-oriented patterns for the business processes. The results of the later research are reported in (Khan, 2012; Ahmed et al., 2012; Ahmed and Matulevičius, 2013). We have applied the security risk-oriented patterns on two (industry) business models modelled in BPMN. Our goal was to understand the early security concerns. This application helped annotate business models with the security related information. We observe that the patterns potentially facilitate non-security experts in seeing the necessity to secure their business processes. Additionally they support the business analysts when making the cost effective decisions, since it enables reasoning on the security risk level and countermeasure cost early at the system development (i.e., business process management) stage. However, it should be also noted that our empirical validation is rather limited and still remains an objective of the future research.

**DISCUSSION**

**Threats to Validity**

Our proposal contains 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, aligned, and extended to the ISSRM concepts differently. Also, the running examples involve the subjective decisions on how problem needs to be modelled. For instance, in majority we have taken risk reduction decisions. 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 the examples discussed in the previous section are realistic, we have not applied them in the practical settings. Thus, our analysis remains based on the selected BPMN literature (White, 2004; Remco et al., 2007; Silver 2009).

**Related Study on Security-oriented BPMN**
The literature suggests few BPMN extensions to model security concerns. For instance Rodríguez et al., (2007a, 2007b) have proposed BPMN extensions for modelling secure business processes through understanding the security requirements. The focus is placed on the perspective of business analysts to help them understand the security concerns. 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, supporting them with graphical icons. They present a symbol of padlock to express security requirements and a padlock with twisted corner for audit register (similarly adopted in our proposal, too). Elsewhere, Menzel et al. (2009) have proposed the BPMN enhancements towards trust modelling. They believe that multiple parameters have to be considered for security configurations. Their proposal is to annotate security intentions and ratings in business processes. The metrics allow giving a value to enterprise assets and concentrating on the level of security, the trust level for each participant in the process. The enterprise assets are presented using BPMN tasks, data objects, and communication links between tasks and participants. In addition, authors define how to enable trustworthy interactions, organisational trust, and security intentions through BPMN. Other proposed extension is a security policy model used to define specific security patterns for authorisation, authentication, integrity, and confidentiality. In comparison to (Rodríguez et al., 2007a; 2007b; Menzel et al., 2009), in this paper we introduce the BPMN extensions based on the fine-grained analysis, i.e., following the previous language alignment (Altuhhova et al., 2012) to the ISSRM domain model. In other words we explore the reasons why and then introduce how BPMN needs to be extended to consider security at the business process modelling.

Elsewhere Mülle et al. (2011), based on the security vocabulary, have introduced the security constraints and security-specific user involvements using BPMN (version 2.0). Each security unit is represented as a structured text annotation, tied to the particular set of BPMN elements (such as tasks, lanes, message flows, etc.). The authors emphasize three security targets: policies, adjustment of the process flow, and parameter settings. The components, then, are used to create relations between users and tasks, applying security-specific policies to the process and helping manage the security concerns. The authors argue that such extensions of the open-source business-process-management system (BPMS) helps transform traditional business process annotated with the security constraints to the executable processes. In our work we introduce the language visual syntax constructs as the part of the modelling language. These constructs (both visual and their semantics) could directly be used for defining security specific policies and managing the security concerns.

Based on the domain for the Information Assurance and Security (IAS), Cherdantseva et al. (2012) have proposed the BPMN extensions that include the systematic management of information security countermeasures and human-oriented aspects. Their major contribution is the representation of the IAS modelling capabilities using BPMN. The authors introduce the alignment table concluding what security extensions are important to introduce and propose the graphical elements to support security modelling. Although work of Cherdantseva et al. (2012) is based on the idea to extend the language based on the semantics, this work focusses on slightly different domain, which targets other security aspects. In our research we specifically concentrate on the security risk management, thus, supporting reasoning for the security requirements.

**Related Study on Security Risk-oriented Modelling Languages**

BPMN is not the only language considered for the IS security risk management. ISSRM has been used to assess KAOS extensions to security (Mayer, 2009), Mal-activity diagrams (Chowdhury et al., 2012), Secure Tropos (Matulevičius et al., 2008b; 2012), and misuse cases (Matulevičius et al.,
But among these, BPMN is the only language originally used to define the business process modelling. We have not found any business modelling language, which would support security analysis; thus the recent standard (White, 2004; Remco et al., 2007; Silver 2009) 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. This analysis opens the way to define (security risk-based) model transformations between these languages. Our results show which language constructs could potentially be transformed to the constructs of another language. Additionally, transformation rules should help identifying (i) what semantics is lost from the models when such a translation is performed; (ii) what semantics needs to be additionally defined in the resulting models; and (iii) what semantics is preserved during the model transformation.

ACKNOWLEDGMENT

This research is partly funded by an ETF grant (contract number ETF8704, Estonian Science Foundation).

REFERENCES


The concrete syntax of the extended BPMN is not dependent on the set of visual constructs proposed in this paper. A different notations could potentially be defined (e.g., using other visual variables (Moody, 2009)) when asset, risk, and risk treatment diagrams are printed in the black/white mode.

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.

The sub-process Use Internet store is considered as business asset since it brings some value – a faster/more convenient way of buying good in comparison to the physical visit to the store. It is also important to notice that at the decomposed levels (e.g., Figure 7, 8, and 14) the single business asset tasks are separately supported by the appropriate IS asset tasks.