UNIVERSITY OF TARTU
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
Institute of Computer Science
Computer Science

Andrei Proskurin

A Comparison of Security Modelling Languages used for Security Risk Management

Bachelor's thesis (6 ECTS)

Supervisor(s): Raimundas Matulevičius

Sven Laur

Tartu 2014

Table of Contents

1 Introduction 5
2 Domain Model 6
Asset 6
Risk 6
Risk Treatment 6
2.2 ISSRM Domain Model 7
2.3 Risk Management 7
2.4 Summary 9
3 Modelling Languages 10
3.1 Business Process Model and Notation (BPMN) 10
3.2 Secure Tropos 12
3.3 Misuse Cases (MUC) 15
3.4 Mal-Activity Diagrams (MAL) 17
3.5 Summary 20
4 Empirical Comparison 21
4.1 Study design 21
4.2 Results 22
Model comprehension 22
Language constructs 23
4.3 Threats to Validity 25
4.4 Summary 25
5 Conclusion 26
6 References 28
Appendix 30
I. Attachements 30
II. License 43

A Comparison of Security Modelling Languages used for Security Risk Management

Abstract

Nowadays, every company that has valuable assets has an urge to protect them. Unfortunately, it is impossible to act on every single security threat. To mitigate these threats Security Modelling Languages were extended to use for Security Risk Management. However, choosing suitable language can be a difficult decision, because it can be a problem to compare those languages and decide which one would bring the most cost-effective solution.

Every security solution has its cost and companies have limited resources. The chosen language that will be used for Security Risk Management must suit the company's needs, as it is important in terms of getting positive ROI (Risk on investment). In addition, Security Risk Management takes place on early stages of IS development and choosing security modelling language that does not suit the company's needs will result in a loss of time as well as possible system vulnerabilities.

Our technical contribution to the solution to this problem is a comparison of these Security Modelling Languages: BPMN, Secure Tropos, Misuse cases and Mal-activity diagrams. It is important to determine how these languages act with Information System Security Risk Management (ISSRM) domain model. The comparison is made based on the case study and empirical research in order to understand the semiotic clarity of these languages used to express the security concerns. The empirical research within the case study will allow us to point out in which ways one language acts better than another regarding ISSRM.

The chosen security modelling languages contain limitations regarding the semiotic clarity, as they were not designed to deal with the security risk management at the first place, but used in terms of ISSRM, they help to mitigate risks starting from early stages of IS development.

Keywords:

Asset, Threat, Modelling Languages, Risk Management, IS development, Vulnerabilities, Domain model, Mitigate

Julgeolekuriskide juhtimisel kasutatavate modelleerimiskeelte võrdlus

Kokkuvõte

Tänapaeval kõik firmad, mis omavad väärtuslikke varasid, püüavad oma aktiva ja pasiva kaitsta. Kahjuks ei ole võimalik reageerida kõikidele varade turvalisust puudutavaid ähvardustele. Selliste võimalike ohtude leevendamiseks olid laiendatud modelleerimiskeeled turvariskide halduse kasutamiseks. Sobiva keele valik võib aga olla keeruline otsus, kuna see on iseenesest ränk küsimus, kuidas need keeled omavahel võrrelda ning otsustada kumb lahendus on rentaabel.

Iga turvateenusel on oma hind, kuigi firmad on oma eelarvega piiratud. Konkreetne valitud keel turvariski haldamiseks peab vastama firma vajadustele, kuna see on tähtis positiivse "ROI" (investeeringu risk) suhtes. Samas turvariski haldus asub infosüsteemi arendamise varajasel staadiumil ja keele valik, mis ei vasta firma vajadustele, võib viita aja kaotusele või isegi süsteemi turvaaukudele.

Selle probleemi lahenduseks on meie tehniline panus võrrelda modelleerimiskeel: "BPMN", "Secure Tropos", "Misuse case" ja "Mal-activity" diagramm. On tähtis määratlema, kuidas need keeled toimivad infosüsteemi turvariskide haldamine (ingl. ISSRM) domeeni mudeliga. Juhtumisel ja empiitilisel analüüsil põhinev võrdlus oli tehtud selleks, et selgust saada turvalisuse probleeme puudutavatest keeltest ja nende semiootilisest selgusest. Empiiriline analüüs juhtumi analüüsiga võimaldab välja selgitada, mismoodi üks keel toimib paremini kui teine "ISSRM" suhtes.

Valitud modeleerimiskeeled turvariskide halduseks on mingil määral piiratud semiootilise selguse suhtes, kuna need pole olnud esialgu mõeldud tegelema turvariskide haldusega, pigem "ISSRM" kasutamiseks ning selleks, et aidata ohud leevendada infosüsteemi arendamise varajasel staadiumil.

Võtmesõnad:

Omand, vara, modeleerimiskeeled, turvariskide haldis., infosüsteemi arendus, infosüsteemi turva aukud, domeeni mudel

1 Introduction

Security risk management (SRM) plays a vital role in modern information system development process. It assists in lowering the possible risks and costs by considering these risk at early stages of development. However, the role of SRM is often overlooked which results in considering risks only during the implementation or maintenance process of an information system [7]. Such action leads to insufficient security level as well as high cost of risk treatment.

Modelling languages (e.g. BPMN [5], Secure Tropos [9], Misuse Cases [9], Mal-Activity Diagrams [10]) assist human stakeholders in terms of SRM. These languages provide means to early identify and treat possible risks which can occur during different stages of development, implementation and maintenance processes.

In this paper we compare security modelling languages regarding comprehension of different language concepts by human stakeholders. Our scope includes BPMN which is widely used for modeling business processes using the graphical presentation of different aspects [4], Secure Tropos which are based on Tropos methodology (use of Actor, Goal, Task, Resource and Dependency concepts) [8], Misuse Cases which as opposed to traditional Use Cases present essentially structured story of system misuse [9], and Mal-Activity Diagrams which main idea is presenting the activities leading to negative impact on a system [10].

Our main research question is: "What of the presented SRM languages is better understood in terms of concepts comprehension [11] and language constructs by the *human stakeholders* than the other?" In order to answer this question we needed to compare the modelling languages. Our first step was to understand the concepts of Information Systems Security Risk Modelling (ISSRM) domain [11]. Afterwards, we used the modelling languages to construct the models on one particular case basis. These models were further given to an audience with a view to understand how human stakeholders comprehend different models and language constructs.

In Chapter 2 we present ISSRM concepts and domain, both used in terms of the comparison basis. Chapter 3 introduces the SRM languages and delivers their models. In Chapter 4 we introduce our study and the results of the collected data analysis. Lastly, in Chapter 5 we discuss the results of the study and make a conclusion.

2 Domain Model

Information Systems Security Risk Management (ISSRM) is an extremely important activity regarding the development of secure systems. Its main goal is to use risk management approach to protect *assets* of an organization from all dangers to IS security which are possibly to occur. Its domain model presents the concepts of ISSRM and how they relate to one another. In this section we summarize some core definitions of ISSRM concepts, organized in three categories: *asset*-related concepts, *risk*-related concepts and *risk-treatment* related concepts [1].

2.1 Core definitions

Asset

Asset-related concepts define which assets of an organization need to be protected. *Assets* are everything that have a value to a company and are important in terms of achieving its goals. A business *asset* introduces valuable information, processes, capabilities and skills important to the business of the organization. An IS *asset* is a part of the information system (IS) supporting business *assets*. *Security criterion* which is defined in terms of business *assets* are usually confidentiality, integrity and availability. However, considering the context, other *criterion* can also be added [1].

Risk

Risk-related concepts describe the *risk* and how exactly it can occur. *Risk* is a combination of *Event* and *Impact* which describe negative effects harming company's assets. *Event* introduces the occasion that had led to the *Impact*. *Event* concept consists of *Threat* and *Vulnerability* concepts. *Vulnerability* is a characteristic of an IS *asset* which shows through which place of a system an attack can be carried out. *Threat* itself is a combination of *Threat agent* and *Attack method* concepts which describe an agent that has an intention to harm organization's assets and an attack technique that the agent is using to exploit the *Vulnerability* [1].

Risk Treatment

Risk treatment-related concepts introduce the decisions to treat identified *Risks*. *Risk treatment* presents methods to mitigate the risks. *Security requirement* is a refinement of a *Risk treatment*

to mitigate the risk. *Control* is designed to improve the security defined by the *Security requirement* and is implemented to work with it [1].

2.2 ISSRM Domain Model

The ISSRM domain model presented in Fig. 1 is a result of an alignment of all three principal groups of ISSRM concepts: (i) asset-related concepts, (ii) risk-related concepts, and (iii) risk treatment-related concepts. The name for each concept as well as the relationships on which the concepts are linked were identified in [11].

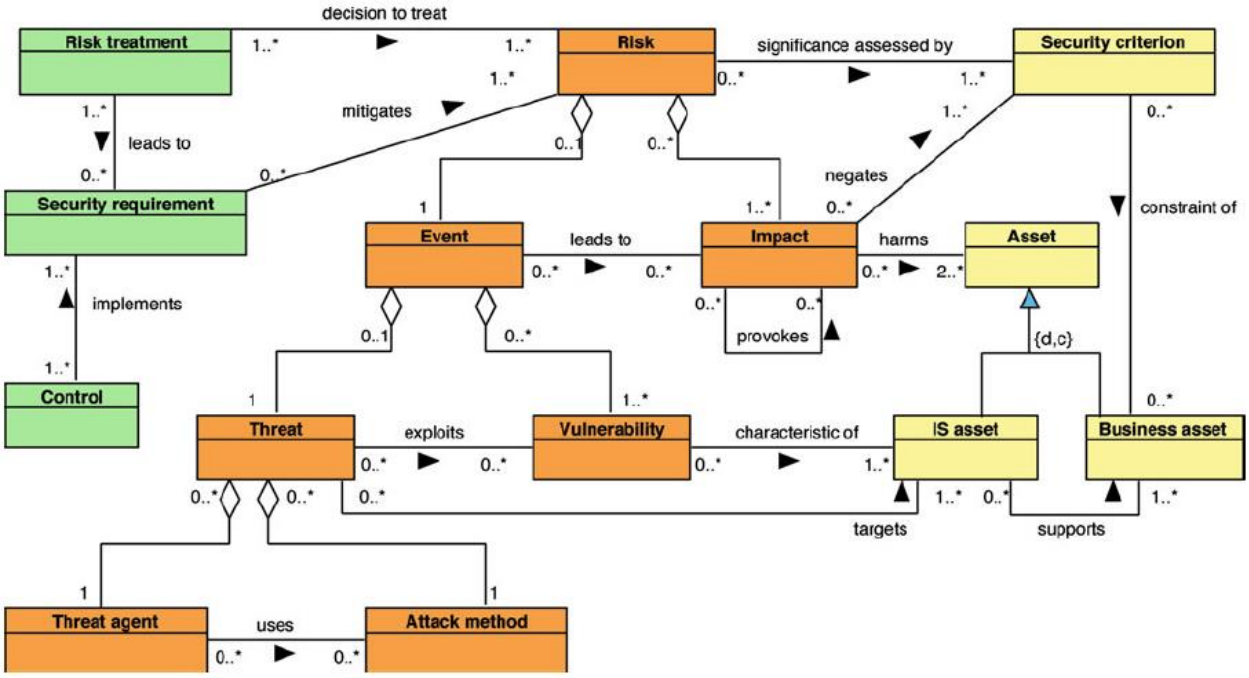


FIGURE 1: ISSRM DOMAIN MODEL [12]

The presented ISSRM domain model highlights the main ISSRM concepts and their relationships, together with their corresponding definitions [11].

2.3 Risk Management

Security Risk Management process is an iterative event and can be performed many times until acceptable level of satisfaction is achieved. The process is shown in Fig. 2. It consists of 6 stages [3]:

Step 1. Context and asset identification

At the start of RM process we need to outline the company's context and valuable *assets*. In this step we need to analyze the organization and the IS in depth.

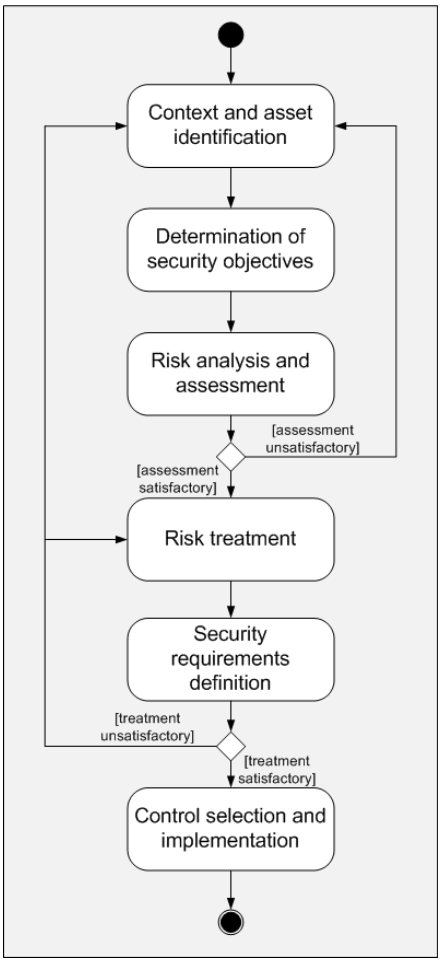


FIGURE 2: RISK MANAGEMENT PROCESS

Step 2. Determination of security objectives

At this stage, security objectives of the organization are defined. These objectives are based on the organization's assets and they are usually described in terms of confidentiality, integrity and availability properties of the assets.

Step 3. Risk analysis and assessment

Risk analysis is the most important step of RM. At this stage the organization's assets as well as the security objectives must be taken into consideration in order to define risks that appear as threats to them. Risk level is also identified. Unless risk assessment is considered as satisfactory, these 3 steps of ISSRM process are performed iteratively (as shown in Fig. 2).

Step 4. Risk treatment

Risk treatment measures can include avoiding, reducing, transferring or retaining risk [3]:

a. Risk avoiding means trying to exclude usage of risky functionality
b. Risk reducing is applying security requirements or any other measures in order to lessen the probability of risk taking place.
c. Transferring risk is sharing the risk consequences with another party.
d. Retaining risk means accepting the risk because its probability is very low or the loss is inconsiderable.

Step 5. Security requirements definition

Security requirements on the IS can thus be determined as security solutions to mitigate the risks [3]. If security requirements are considered as unsatisfactory, risk treatment step or all the preceding steps should be repeated.

Step 6. Control selection and implementation

At this point security requirements are realized as real solutions to mitigate risks (e.g., firewalls, security systems). As mentioned above, ISSRM process is iterative and should be repeated unless risk treatment is considered as satisfactory.

2.4 Summary

In this chapter we overviewed the main concepts of Information Systems Security Risk Management. We defined its core definitions regarding *asset*-, *risk*- and *risk treatment*-related concepts. In addition to it we presented its domain model with all three groups of concepts linked to each other (Fig. 1) and presented Risk Management process (Fig. 2) including its executable steps.

In the next chapter we will introduce the concepts and syntax of four security modelling languages (BPMN, Secure Tropos, Misuse Cases, Mal-Activity Diagrams) which we will use to create the models based on which the comparison will be carried out.

3 Modelling Languages

In this chapter we present four security modelling languages that are used for security risk management: Business Process Model and Notation (BPMN), Secure Tropos, Misuse Cases and Mal-Activity Diagrams. We will define the core concepts of these languages, outline the general purpose of each of them and see how they could be applied in terms of ISSRM regarding one particular case.

3.1 Business Process Model and Notation (BPMN)

Business Process Model and Notation (BPMN) is a standard for business process modeling that provides a graphical notation for specifying business processes in a Business Process Diagram (BPD) [5]. BPMN is an essential part of IS development, as it helps to specify standard and optimized workflows of organization. The primary purpose of BPMN is modelling of the business processes for both technical users and business users, by providing a notation that is intuitive to business users, yet able to represent complex process semantics. The notation must be understandable by all by all business stakeholders (e.g., analysts, managers). BPMN acts as a bridge between business process design and implementation.

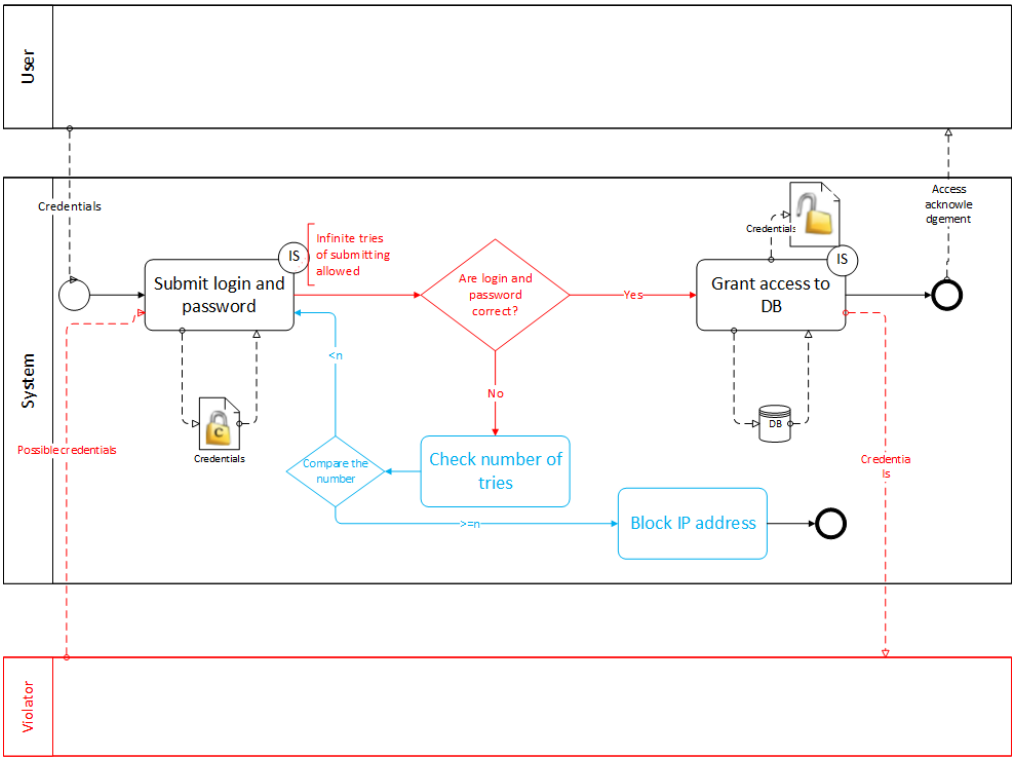


FIGURE 3: BPMN COMBINED MODEL

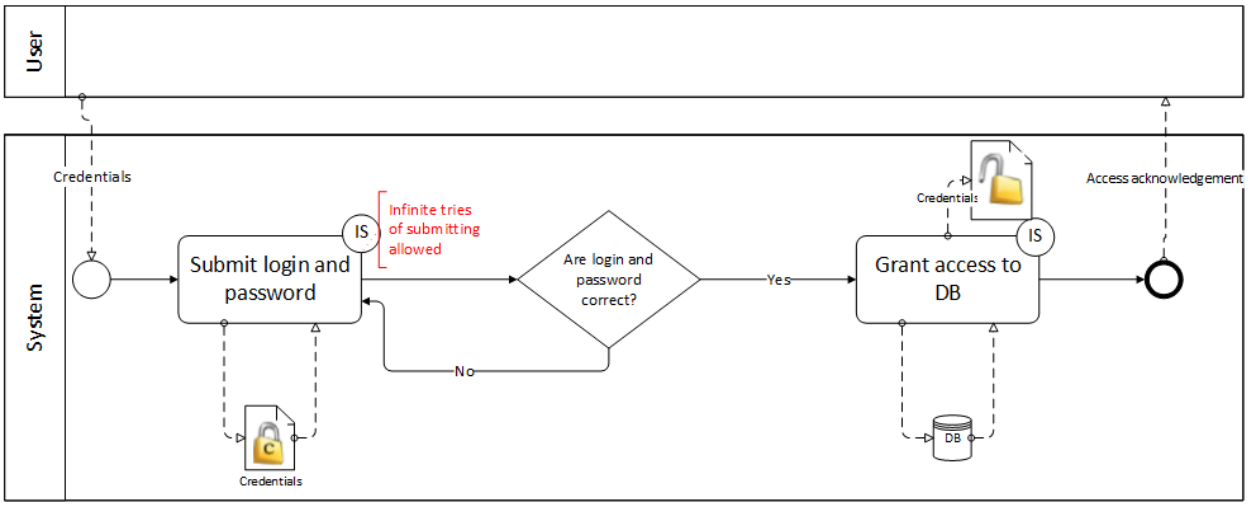


FIGURE 4: BPMN - *ASSET*-RELATED CONCEPTS

Figure 4: The business *asset* is the most valuable *asset* that has to be secured at any cost. In BPMN it is defined using *tasks* and *data objects* (see **Credentials**). As we already mentioned, the business asset needs to stay secure, therefore, the *confidentiality* of **Credentials** is also considered (see *lock* icon at **Credentials**). Business *asset* is supported by IS *assets* that are defined using *Pool* (see **System**) and *Tasks* (see **Submit login and password** and **Grant access to DB**).

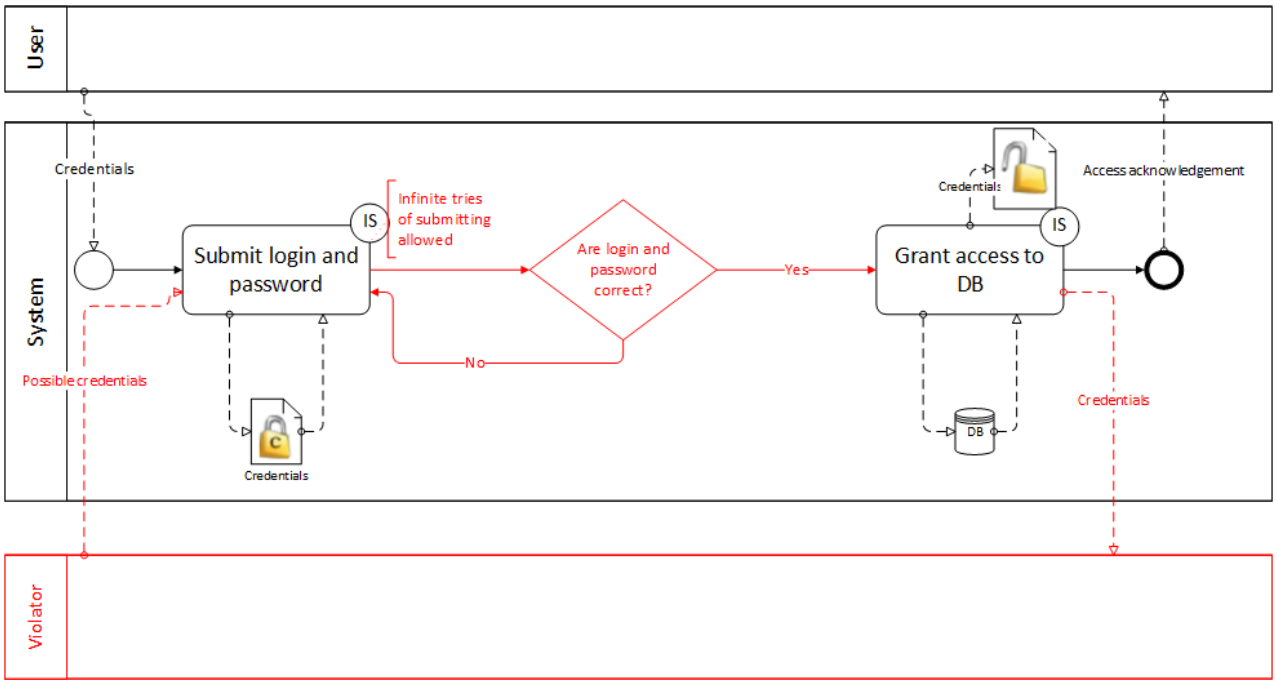


FIGURE 5: BPMN - *RISK*-RELATED CONCEPTS

Figure 5: Security *threat* is defined using BPMN structure for *threat agent* (see *Pool* **Violator**) plus message flow that follows the *task* (see **Possible credentials -> Submit login and password**). The risk *event* is possible because of a security *vulnerability* that is defined using BPMN *Text annotation* (see **Infinite tries of submitting allowed**). The main problem is that when a person submits invalid login and password the system allows him/her to repeat this step infinitely. This results in **Violator** getting access to the database (see *task* **Grant access to DB**). In this particular situation the confidentiality of our business *asset* is negated (see *unlocked* icon at *data object* **Credentials**).

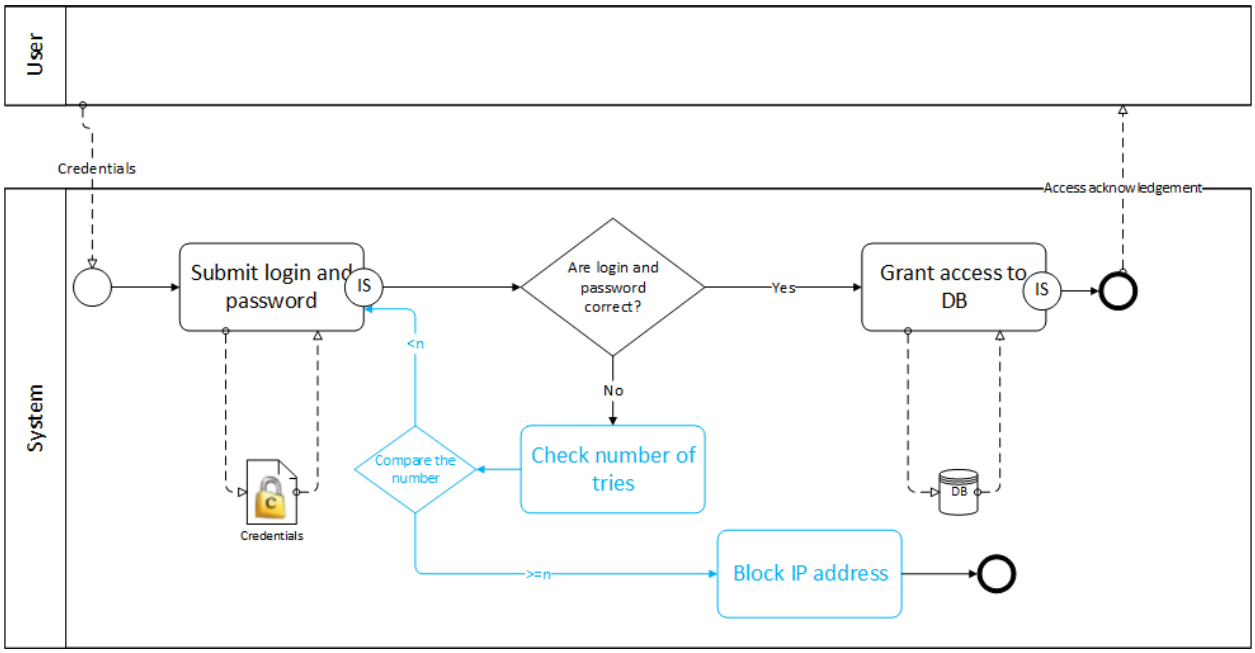


FIGURE 6: BPMN - *RISK TREATMENT*-RELATED CONCEPTS

Figure 6: *Security criterion* (a way to satisfy the security needs the way that reduce the chances of risk *event* happening) is presented using a combination of such BPMN constructs as *Task* (see **Check number of tries** and **Block IP address**) and *Gateway* (see **Compare the number**). It mitigates our *risk* that confidentiality of credentials can be broken and prevents *threat agent* from using the *vulnerability* that infinite tries of submitting allowed.

3.2 Secure Tropos

Secure Tropos is based on the Tropos methodology, which uses the concepts of actor (entity that has strategic goals and intentionality), goal (an actor's strategic interest), soft-goal (goal without

clear criteria whether it is satisfied or not), task (it represents the way of doing something), resource (it represents a physical or informational entity, without intentionality) and social dependencies (indicate that one actor depends on another in order to attain some goals, execute some tasks, or deliver a resource) [8]. It extends the Tropos methodology by adding security concerns during the development process. In particular, Secure Tropos extends the Tropos language as well as its development process. The language extension consists of redefining existing concepts with security in mind as well as introducing new concepts.

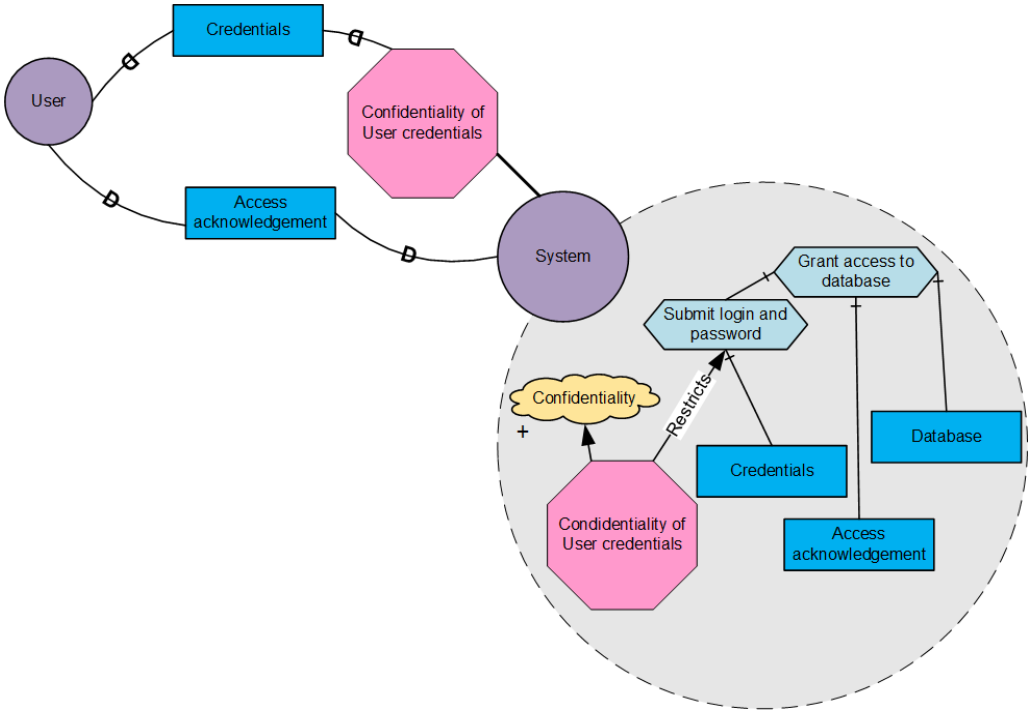


FIGURE 7: SECURE TROPOS - *ASSET*-RELATED CONCEPTS

Figure 7: Development of the Secure Tropos model starts with the social actor dependency analysis. In this figure two dependencies are defined between *actors* **User** and the **System**. A *Security criterion* (e.g. Confidentiality of User credentials) restricts **Submit login and password** and contributes to the general confidentiality property of the systems. To support the *business asset* (e.g. **Credentials**), the *IS assets* are defined using the *actor* (e.g. System) and *plan* (e.g. **Submit login and password**, **Grant access to database**) constructs. In order to **Grant access to database** few *resources* need to be available (e.g. **Credentials** for the **Submit login and password** *plan*, **Access acknowledgement**, **Database**).

Figure 8: To reach his *goal* **User credentials** received, a *threat agent* (e.g. **Violator**) performs the *attack method* (e.g. **Submit login and password**). In this way the attacker exploits the *vulnerability points* (characteristic of plans **Submit login and password**, **Grant access to database**) and targets (e.g. attacks) the **Credentials**.

Figure 9: To mitigate this security *event*, *plan* **Block IP address** is introduced. *Plans* **Check login and password** and **Check number of submits** are part of **Compare to given number (n)** which is itself part of **Block IP address** process. All this together reach the *goal* **User credentials secured** which satisfies **Confidentiality of user credentials**. Satisfied *security criterion* mitigates the *risk* (e.g. **Stealing user credentials**).

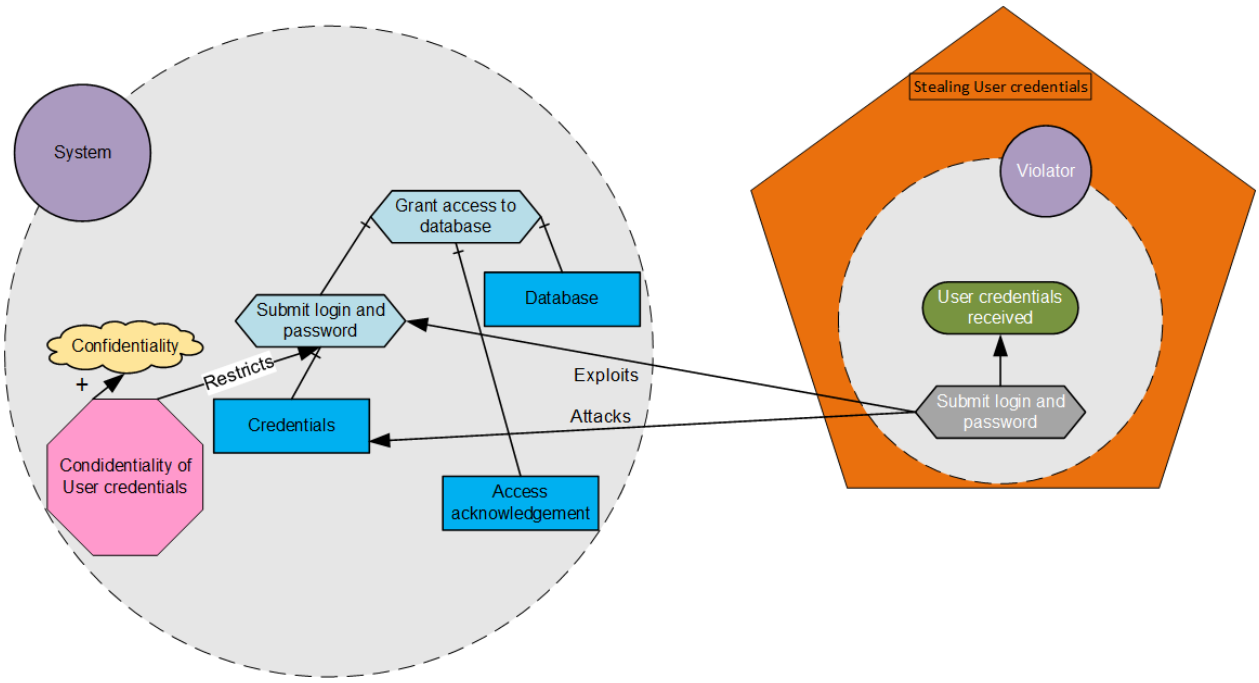


FIGURE 8: SECURE TROPOS - *RISK*-RELATED CONCEPTS

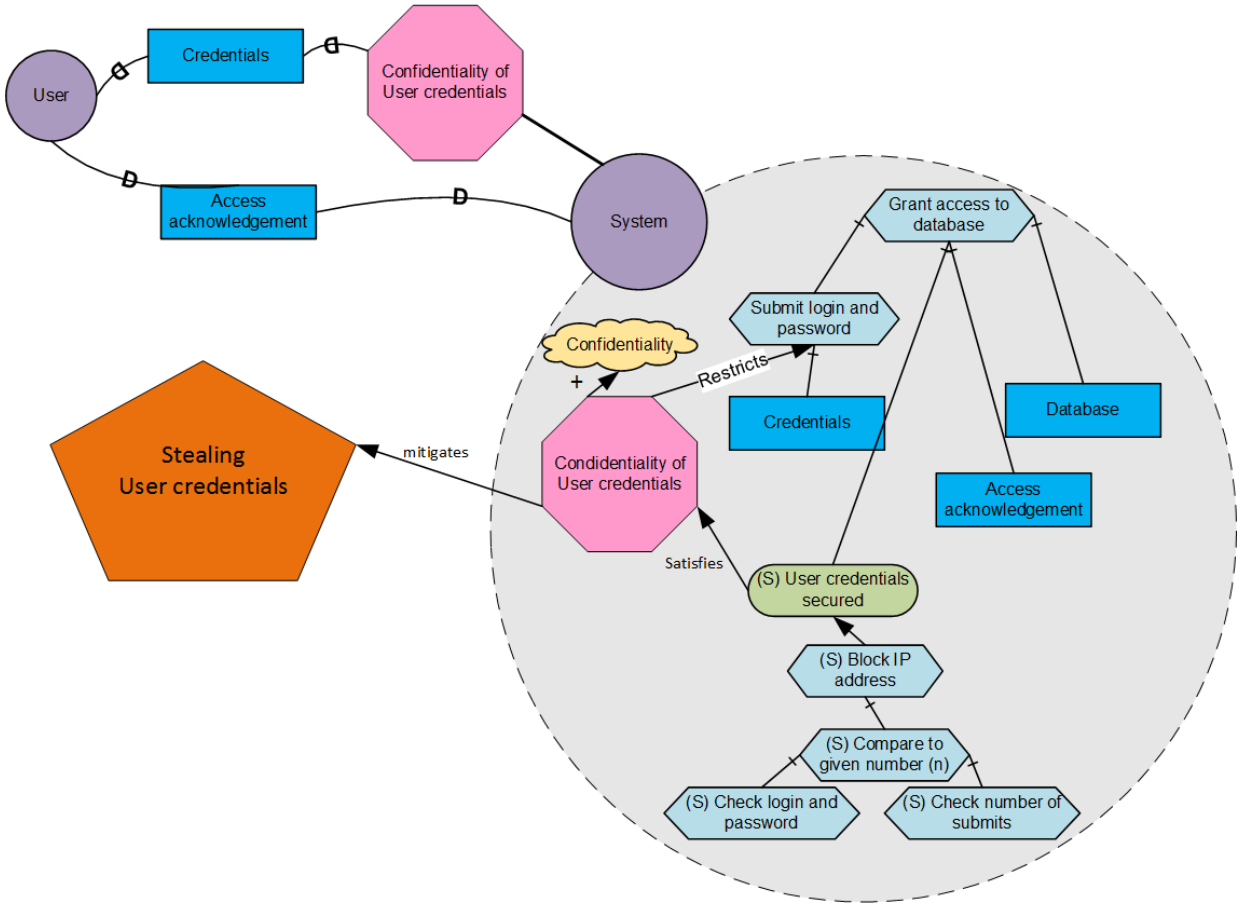


FIGURE 9: SECURE TROPOS - RISK TREATMENT-RELATED CONCEPTS

3.3 Misuse Cases (MUC)

Misuse Case is a business process-modeling tool used in the software development industry. The term Misuse Case or misuse case is derived from and is the inverse of use case. Use case's specify required behavior of software and other products under development, and are essentially structured stories or scenarios detailing the normal behavior and usage of the software. A Misuse Case on the other hand highlights something that should not happen (i.e. a Negative Scenario) and the threats hence identified, help in defining new requirements, which are expressed as new Use Cases [9].

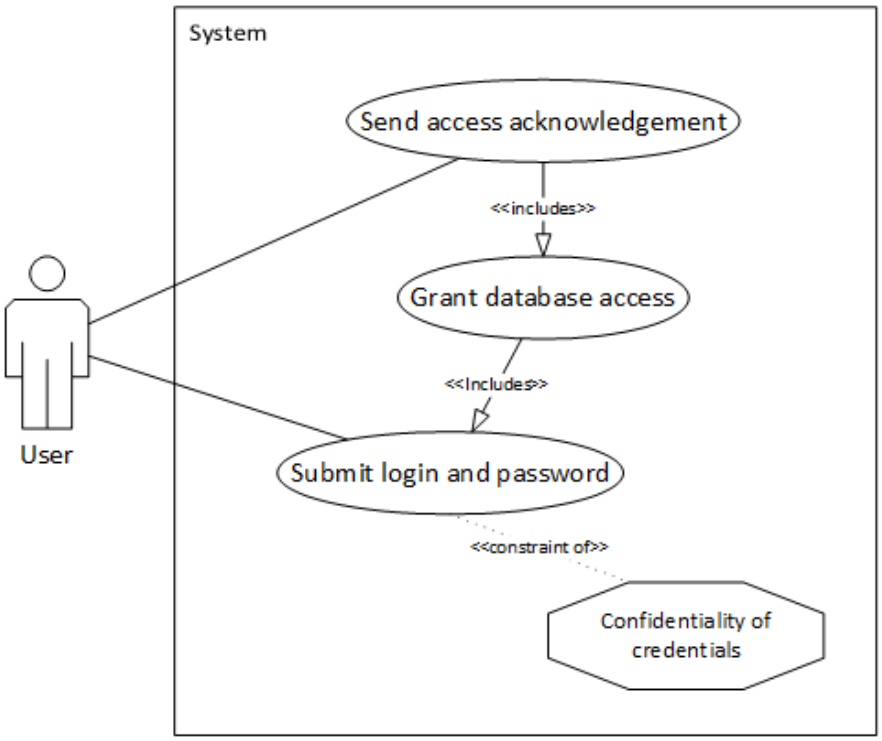


FIGURE 10: MISUSE CASES - *ASSET*-RELATED CONCEPTS

Figure 10: In this figure *actor* **User** communicates to the **System** for **Submit login and password**. The *security criterion* **Confidentiality of credentials** is introduced as a constraint of **Submit login and password**. The *use case* **Send access acknowledgement** includes *use case* **Grant database access** that is understood as *IS asset*.

Figure 11: The *misuser* (e.g. **Violator**) uses a *threat method* (e.g. *misuse case* **Submit login and password**). The *IS asset* **Grant database access** is threatened because it has a *vulnerability* that **Infinite number of submits allowed**. The security *event* leads to the *impact*, which is defined as **User credentials are stolen**. This *impact* harms the *business asset* (e.g. **Submit login and password**) and negates its security *criterion* (e.g. **Confidentiality of credentials**).

Figure 12: To mitigate the identified system misuse, a security *use case* – **Block IP address** is introduced. It consists of the *use case* **Compare to given number (n)** which itself includes *use cases* **Check login and password** and **Check number of submits**. The mitigation is carried on through the *misuse case* **Submit login and password**.

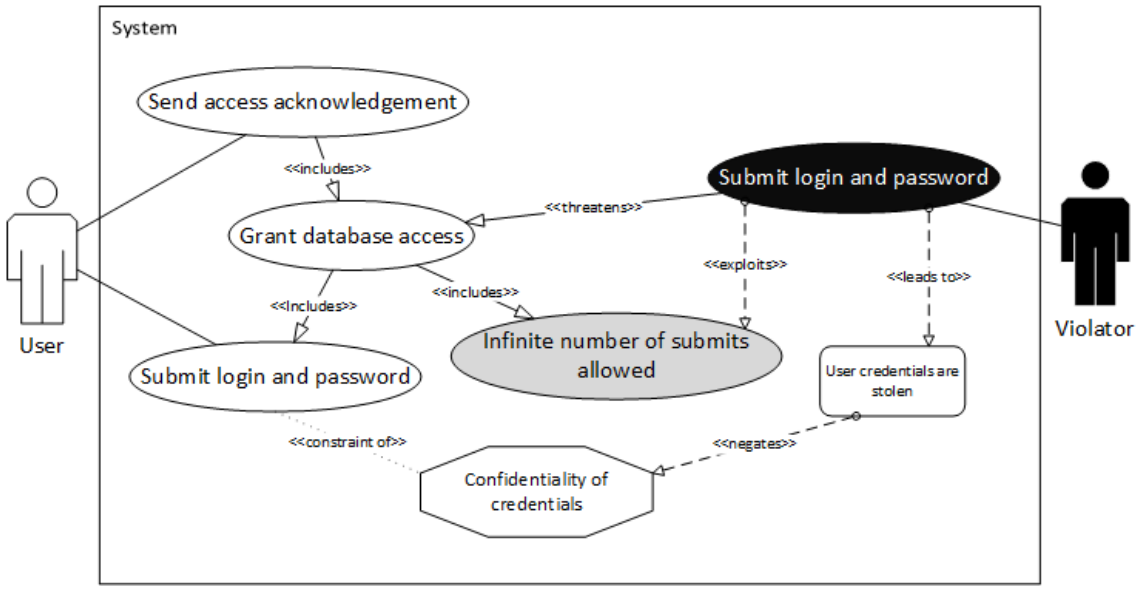


FIGURE 11: MISUSE CASES - *RISK*-RELATED CONCEPTS

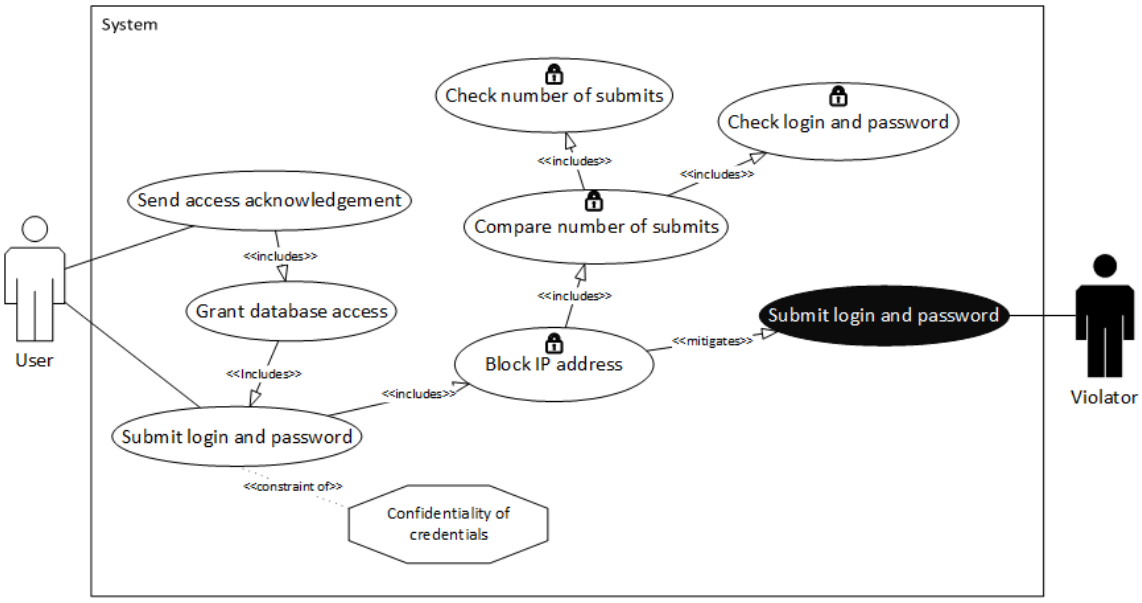


FIGURE 12: MISUSE CASES - *RISK TREATMENT-RELATED CONCEPTS*

3.4 Mal-Activity Diagrams (MAL)

Mal-activity diagrams are proposed as an extension of the UML activity diagrams to model a harmful behavior of security attackers. A basic way to build a mal-activity diagram is to model a normal process first, then to add unwanted behavior using mal-activities, mal-swimlane and mal-decision constructs [10].

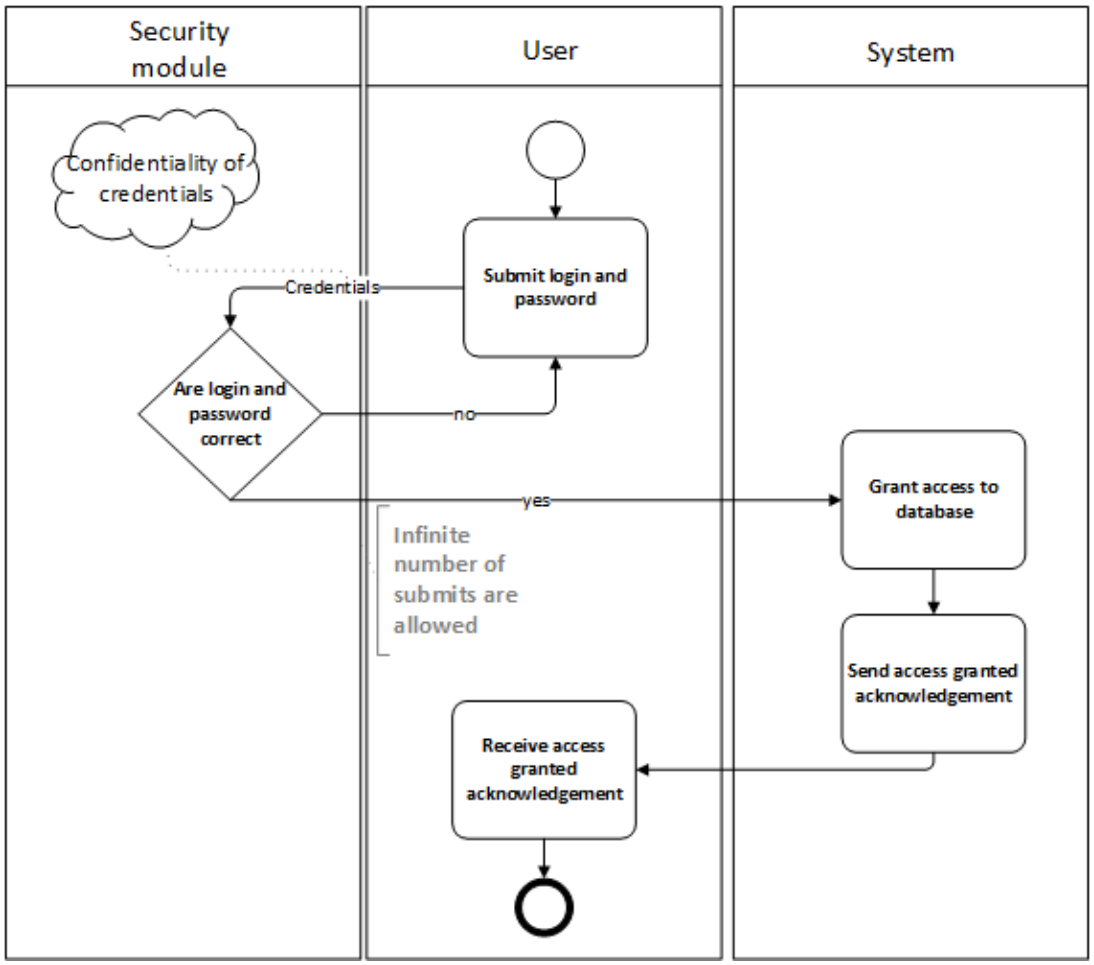


FIGURE 13: MAL-ACTIVITY DIAGRAMS - *ASSET*-RELATED CONCEPTS

Figure 13: In this figure *activities* (e.g. **Submit login and password**, **Receive access granted acknowledgement**) performed under the **User** *swimlane* could be understood as the valuable *business asset*. This process is supported by the *IS assets* characterized as *swimlane* **System** including *activities* **Grant access to database**, **Send access granted acknowledgement** and *swimlane* **Security Module** including *gateway* **Are login and password correct**. During this login process the **Credentials** are submitted to the **Security module** and then passed to the **System**.

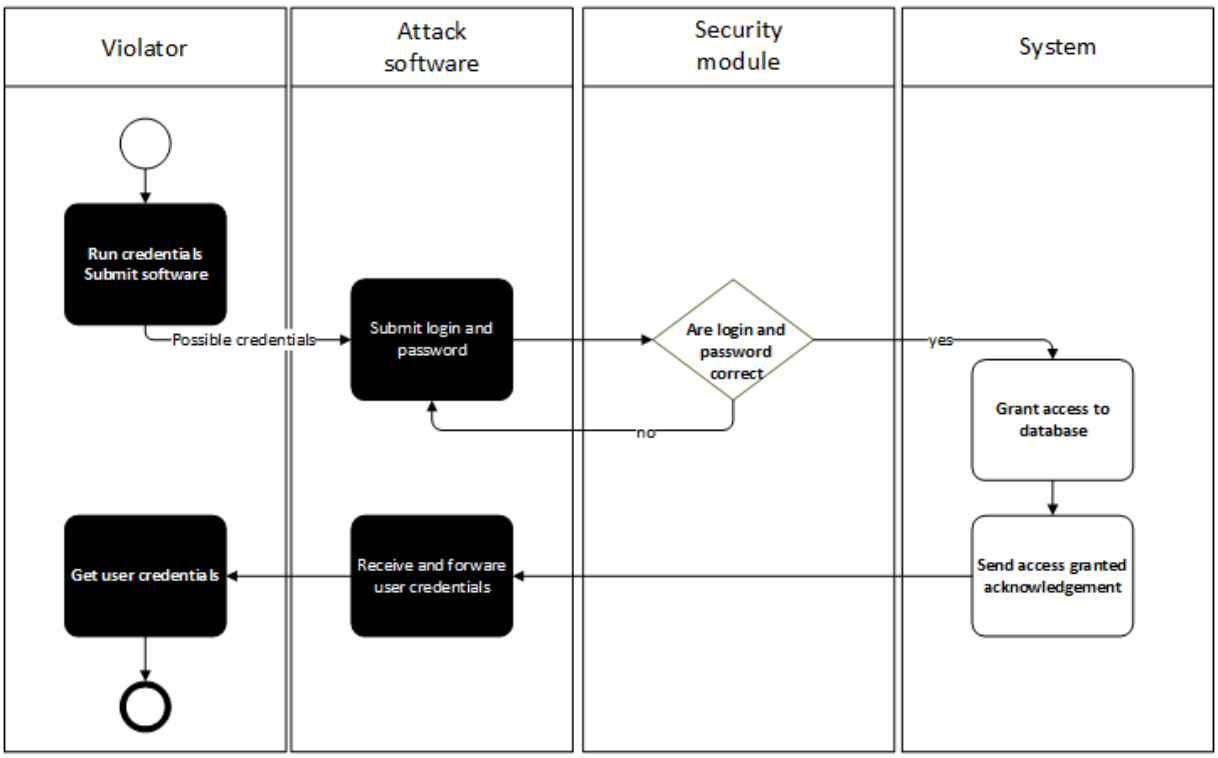


FIGURE 14: MAL-ACTIVITY DIAGRAMS - *RISK*-RELATED CONCEPTS

Figure 14: In order to receive User credentials, the *threat agent* (e.g. *swimlane* Violator) uses credentials submit software (e.g. the attack software) as the means to execute attack. If the *event* is successful, it leads to the *impact*, such that the login and password were submitted (e.g. harm to the IS asset) and the credentials are captured and sent to the *violator* (e.g. harm to the *business asset*). These two aspects of the impact are expressed using the mal-activities contained in the Attack software *swimlane*. By sending the credentials to the *Violator*, Attack software also negates the Confidentiality of credentials.

Figure 15: The *security requirements* Check number of submits and Block IP address are introduced to mitigate the identified *risk*. Check number of submits is introduced immediately after the Are login and password correct and has two possible outcomes. So does the first one. Negative outcome returns to *activity* Submit login and password and positive – to *security requirement* Block IP address. This *security requirement* is defined in the *swimlane* Security module that corresponds to a *security control*.

3.5 Summary

In this chapter we introduced four security modeling languages: BPMN, Secure Tropos, Misuse cases and Mal-activity diagrams. We have observed constructs of each of these languages applied to one particular case study of a random login process regarding *asset*, *risk* and *risk-treatment* concepts. Moreover we pointed out how differently could four languages express the same aspect (e.g. Actor, Threat, Security requirements etc.).

In the next chapter we will introduce the empirical comparison on these four languages that has been carried out. We will define the goal of the study, introduce the audience, describe the process of the study and, moreover, present the results.

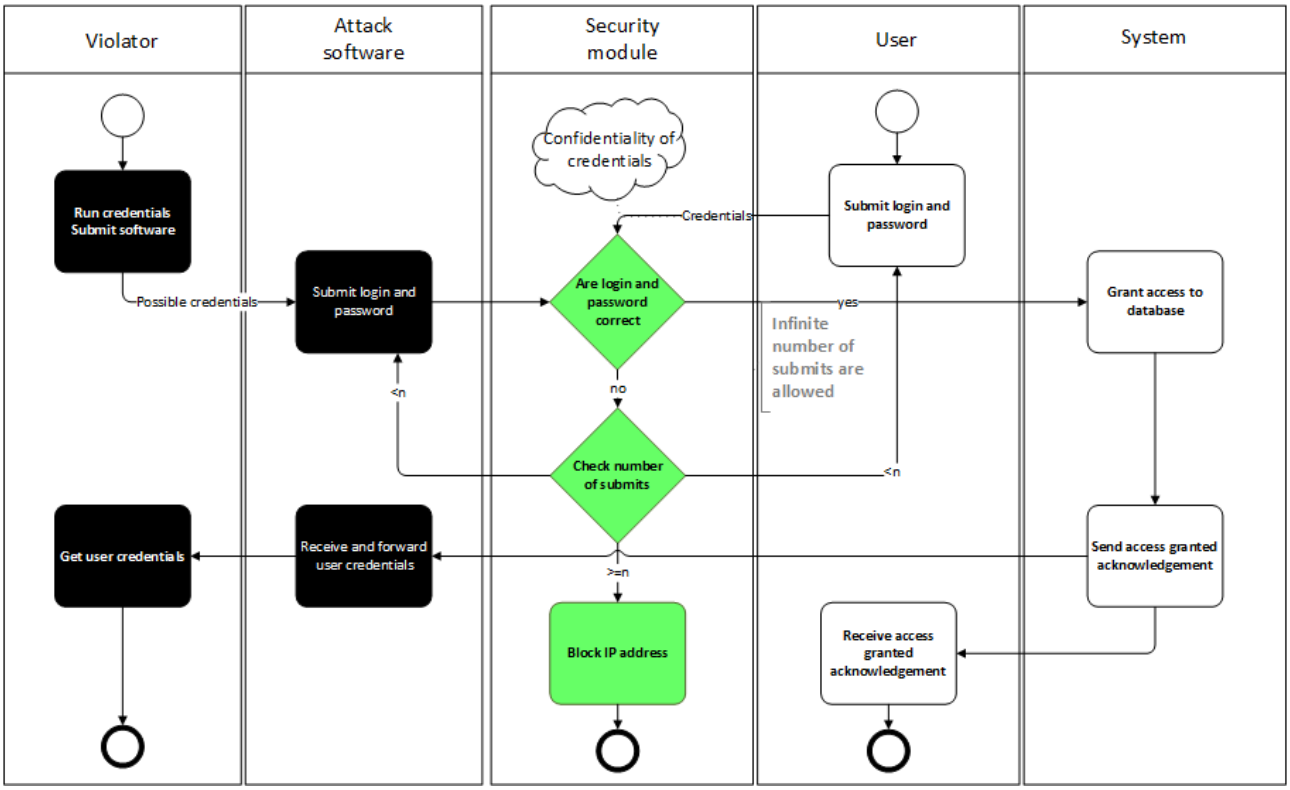


FIGURE 15: MAL-ACTIVITY DIAGRAMS - *RISK TREATMENT*-RELATED CONCEPTS

4 Empirical Comparison

4.1 Study design

In order to answer our research question we needed to evaluate how *human stakeholders* perceive the IS security risk modelling languages presented in Chapter 3. To complete this task we have conducted an empirical study within the University of Tartu. Our scope included 48 students attending the course "Principles of Secure Software Design" [12] at year 2014 (case 1) and also 39 students attending the same course at year 2013 (case 2). All of them were graduate (1st year of Master's degree, overall 4th year of university curriculum).

The course included lectures and practice sessions on principles of the security risk management, risk modelling, security requirements, model driven security, and development processes of the secure software. Supporting and explaining reading material was also presented to the students, as well as main principles of risk analysis and assessment, and the ISSRM domain model [7].

The models based on the case study were then presented to the students. The models of each modelling language included 4 different diagrams, which consisted of asset-related concepts, risk-related concepts, risk treatment-related concepts, and all 3 combined. Each 4 models of every introduced language were followed by the questionnaire. The study included 2 major phases. In first phase, students were asked to analyze the presented models and indicate the exact ISSRM concepts expressed there and fill in the 1st part of the questionnaire. For example, the business asset was mostly described by Account Data field or related to it. Some aspects (e.g. IS asset) were described in the models using a combination of constructs. For instance IS asset in BPMN model was described by Submit login and password and Grant access to DB tasks. There is also the case that given models did not represent the Control concept and the survey participants had to write "not presented" in this particular example.

In 2nd phase of the study, participants had to define the exact language constructs that were used to describe the asset, risk and risk treatment concepts. This phase assisted us in evaluating the comprehension of the SRM languages, as this part gave as understanding of whether the students fully understand the principles and legend of given languages. For example, the threat agent concept in Secure Tropos diagram was described using Actor construct. If some concept was described using a combination of constructs, all of them had to be specified in order to count it as

a correct answer (e.g. a Threat concept in Misuse case diagram was described using a combination of Misuser and Misuse case constructs).

4.2 Results

The total number of questionnaires filled was:

- 20 for Business Processes Model and Notation diagrams
- 18 for Secure Tropos diagrams
- 24 for Misuse case diagrams
- 24 for Mal-activity diagrams

We have excluded the Control score for the Security Tropos model when calculating the total score for case 1 due its low value (0%).

Model comprehension

In Table 1 we present the results for the comprehension of the models which we created using the SRM languages. The score is calculated as the percentage of the correct replies left by the respondents. For example, the IS asset concept in Asset concepts group for Secure Tropos is 80% which means that only 80% of respondents (8 out of 10 in this particular case) correctly defined the IS concept (or a combination of them) in Secure Tropos model. For the overall score we calculated the average value for the exact row/column. The percentage written in blue indicates the global score for the percentage of correct replies regarding the model comprehension aspect. It is calculated as the average of the **Overall** column.

Regarding the 1st case, Table 1 indicates that eight ISSRM concepts (*IS asset, Risk, Event, Threat, Threat Agent, Attack Method, Security Requirement, Control*) were understood better from Mal-Activity diagrams, two concepts (*Security Criterion, Vulnerability*) from Misuse Cases, two concepts (*Business asset, Impact*) from Secure Tropos and one concept (*Risk Treatment*) from BPMN. In terms of model scores, the best perceived model was the Mal-Activity Diagram which overall score for the model comprehension is 74%. According to the case 2 data, we can point out that the best perceived was the Secure Tropos model which overall score for the model comprehension was 79%.

TABLE 1: COMPREHENSION OF THE MODELS

ISSRM Concepts		Security risk-oriented				Overall 2014 Case 1	Overall 2013 Case 2
		BPMN model	Secure Tropos model	Misuse case diagram	Mal-activity diagram		
Number of valid responses		13	10	14	10	47	39
Asset concepts	Business asset	62%	80%	57%	60%	**65%**	**85%**
	IS asset	92%	80%	64%	100%	**84%**	**64%**
	Security criterion	77%	90%	100%	90%	**89%**	**94%**
Risk concepts	Risk	38%	20%	43%	60%	**40%**	**47%**
	Impact	85%	90%	71%	80%	**81%**	**75%**
	Event	46%	40%	57%	70%	**53%**	**60%**
	Vulnerability	62%	40%	79%	60%	**60%**	**61%**
	Threat	38%	50%	64%	80%	**58%**	**53%**
	Threat agent	85%	80%	86%	90%	**85%**	**94%**
	Attack method	46%	60%	57%	80%	**61%**	**69%**
Risk treatment concepts	Risk treatment	77%	50%	71%	70%	**67%**	**54%**
	Security requirement	46%	50%	64%	70%	**58%**	**54%**
	Control	23%	*0%*	14%	50%	**29%**	**7%**
Overall 2014 (Case 1)		**57%**	**61%**	**64%**	**74%**	**64%**	-
Overall 2013 (Case 2)		**74%**	**79%**	**63%**	**54%**	-	**63%**

In terms of the exact ISSRM concepts for the both cases the best understood ones were *Security criterion* and *Threat agent* (case 1 – 89% and 85%, case 2 – 94% and 94%). The concept which received the lowest score was *Control* for the both cases (29% and 7% respectively). The global scores regarding the model comprehension aspect were relatively the same for both cases (case 1 – 64%, case 2 – 63%).

Language constructs

In Table 2 we introduce the results for the understanding of the language constructs of the SRM languages which were used to constructs the models. The score is calculated as the percentage of construct definitions written correctly by the respondents. The IS asset concept construct in Asset concepts group for Secure Tropos is 70% which means that 70% of respondents (7 out of 10 in this case) correctly defined what language construct (or combination of them) was used to express IS asset in Secure Tropos model. For the overall score we calculated the average value

for the exact row/column. The percentage written in blue indicates the global score for the percentage of correct replies regarding the language constructs aspect. It is calculated as the average of the Overall column.

TABLE 2: LANGUAGE CONSTRUCTS

ISSRM Concepts		**Security risk-oriented**				**Overall 2014 Case 1**	**Overall 2013 Case 2**
		BPMN model	**Secure Tropos model**	**Misuse case diagram**	**Mal-activity diagram**		
Number of valid responses		13	10	14	10	47	39
Asset concepts	Business asset	46%	80%	36%	30%	**48%**	**46%**
	IS asset	77%	70%	43%	50%	**60%**	**30%**
	Security criterion	15%	80%	79%	50%	**56%**	**36%**
Risk concepts	Risk	46%	10%	43%	30%	**32%**	**33%**
	Impact	46%	20%	57%	50%	**43%**	**24%**
	Event	54%	30%	21%	20%	**31%**	**19%**
	Vulnerability	54%	20%	71%	20%	**41%**	**41%**
	Threat	38%	30%	57%	20%	**36%**	**35%**
	Threat agent	77%	70%	64%	40%	**63%**	**53%**
	Attack method	62%	30%	71%	40%	**51%**	**28%**
Risk treatment concepts	Risk treatment	38%	10%	43%	30%	**30%**	**37%**
	Security requirement	62%	40%	43%	50%	**49%**	**32%**
	Control	38%	*0%*	29%	30%	**32%**	**29%**
Overall 2014 (Case 1)		**50%**	**41%**	**51%**	**35%**	**44%**	**-**
Overall 2013 (Case 2)		**52%**	**24%**	**47%**	**9%**	**-**	**34%**

Regarding the 1st case, Table 2 reveals that language constructs for six ISSRM concepts (IS asset, Risk, Event, Threat agent, Security requirement, Control) were better understood by respondents reviewing BPMN model, constructs for two concepts (Business asset, Security criterion) reviewing Secure Tropos model, and constructs for five concepts (Impact, Vulnerability, Threat, Attack method, Risk treatment) reviewing Misuse Case model. In terms of model scores, we can point out that for the both cases the best perceived models regarding their language constructs are the Misuse Case diagram and BPMN model (case 1 – 51% and 50%, case 2 – 47% and 52%). Threat agent language construct was understood better than the others

regarding both case 1 and case 2 (63% and 53% respectively). The concept which received the lowest scores were Risk treatment for the 1st case (30%) and Event for the 2nd case (19%). The global scores regarding the language constructs aspect are 44% for case 1 and 34% for case 2.

4.3 Threats to Validity

One possible threat could be an insufficient knowledge of the four SRM languages which were used to create the models. In order to mitigate, we provided respondents with a study material including principles of ISSRM domain model, the domain model itself (introduce in Fig. 1), and also the basics of RM process (as shown in Fig. 2). Another threat to validity is that participants had lack of motivation to assess the given SRM languages and their models. To mitigate, they were rewarded with the subject points, however, of a small amount [7].

4.4 Summary

In this chapter we presented the empirical research that has been conducted within the University of Tartu. We also introduced the study design as well as the audience. Moreover, we demonstrated the research results which were achieved by analyzing the collected data. Finally, we outlined possible threats to validity.

In the last chapter we will conclude our research by answering our main research question. In addition to it we will discuss the limitations and possible future work which could be done regarding our research.

5 Conclusion

In this paper we introduced principles of SRM as well as four SRM languages – BPMN [5], Secure Tropos [8], Misuse Cases [9], Mal-Activity Diagrams [10]. We used them to create the models which were forwarded to the audience in order to understand how *human stakeholders* perceive the ISSRM concepts and the SRM language constructs. We conclude our study by answering our main research question.

During our research we have faced limitations that had impact on quality of the results as well as ability to answer the research question. Firstly, we received only 86 replies which is insignificant amount and cannot give the exactly correct answer for our research question. In addition to it, some of the replies were not filled appropriately so it was difficult to outline whether it was done on purpose or not.

All in all, the best understood models in terms of comprehension are Mal-Activity Diagram (overall score 74%) for the 1st case and Secure Tropos model (overall score 79%) for the 2nd case. However, the difference between Mal-Activity Diagram and Misuse Case (case 1) is only 10% when they had different number of respondents (10 and 14 respectively). We can also point out that the best understood model regarding both case 1 and case 2 is Secure Tropos model (average score for both cases – 70%). The best perceived ISSRM concepts are *Security criterion* and *Threat agent* (both cases).

We have found that the best perceived model in terms of SRM language constructs was the Misuse Case (case 1 - overall score 51%). However, the difference between Misuse case diagrams and BPMN model is only 1% which cannot be counted as significant distinction. Moreover, Misuse Case and BPMN have the highest score in terms of language constructs aspect regarding both case 1 and case 2. In relation to ISSRM concepts, *Threat agent* language construct is understood better than the other concerning both cases (63% and 53%).

Such results can be explained by the fact that some concepts (e.g. *Threat agent*, *IS asset*) are relatively simple regarding their definitions compared to composite concepts (e.g. *Threat*, *Risk)* that appear to be a combination of two or more other concepts. It outlines the limitations of the languages which were not initially designed, but extended to comply with terms of SRM.

As future work, we plan to expand the study by collecting and analyzing more data regarding the model comprehension and language constructs. We need to proceed to a deeper analysis of the languages that appeared to be better than the other in terms of model comprehension and language construct aspects. We also need to create a tool capable of transforming a model created using one SRM language to another.

6 References

[1] Mayer, N.; Dubois, E.; Matulevičius, R.; Heymans, P. (2008). Towards a measurement framework for security risk management.In: Proceedings of the Workshop on Modeling Security (MODSEC08) held as part of the MODELS 2008: Proceedings of the Workshop on Modeling Security (MODSEC08) held as part of the MODELS 2008. , 2008.

[2] N. Mayer, A. Rifaut, and E. Dubois, "Towards a Risk-Based Security Requirements Engineering Framework", 11th International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'05), in conjunction with the 17th Conference on Advanced Information Systems Engineering (CAiSE'05), Porto, Portugal, June 2005. Mayer, N.: Model-based Management of Information System Security Risk, University of Namur, April 2009.

[3] N. Mayer, "Model-based Management of Information System Security Risk", University of Namur, April 2009.

[4] Matulevicius, R. (2012). Comparing Modelling Languages for Information Systems Security Risk Management. Seyff, N.; Koziolek, A. (Toim.). Modelling and Quality in Requirements Engineering (197 - 210).Verlagshaus Monsenstein und Vannerdat.

[5] Altuhhova, O.; Matulevičius, R.; Ahmed, N. (2012). Towards Definition of Secure Business Process. In: Lecture Notes in Business Information Research: CAiSE 2012 International Workshops, Workshop on Information Systems Security Engineering. (Toim.) Bajec, M.; Eder, J., Springer Heidelberg, 2012, (Lecture Notes in Business Information Research), 1 - 15.

[6] K. J. Higgins, Hacker's Choice: Top Six Database Attacks, http://darkreading.com, May 2008.

[7] R. Matulevicius, "Model comprehension and state appropriateness of security risks of modelling languages", accepted at EMMSAD 2014.

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

[9] Soomro, Inam; Ahmed, Naved. Towards Security Risk-Oriented Misuse Cases. Business Process Management Workshops, Springer Berlin Heidelberg, 132(132), 689 - 700, 2013.

[10] Guttorm Sindre: Mal-Activity Diagrams for Capturing Attacks on Business Processes. REFSQ 2007: 355-366, Department of Computer and Information Science, Norwegian University of Science and Technology, NO-7491 Trondheim, Norway.

[11] E. Dubois, P. Heymans, N. Mayer, R. Matulevicius, "A Systematic Approach to Define the Domain of Information System Security Risk Management", Intentional Perspectives on Information Systems Engineering, 2010.

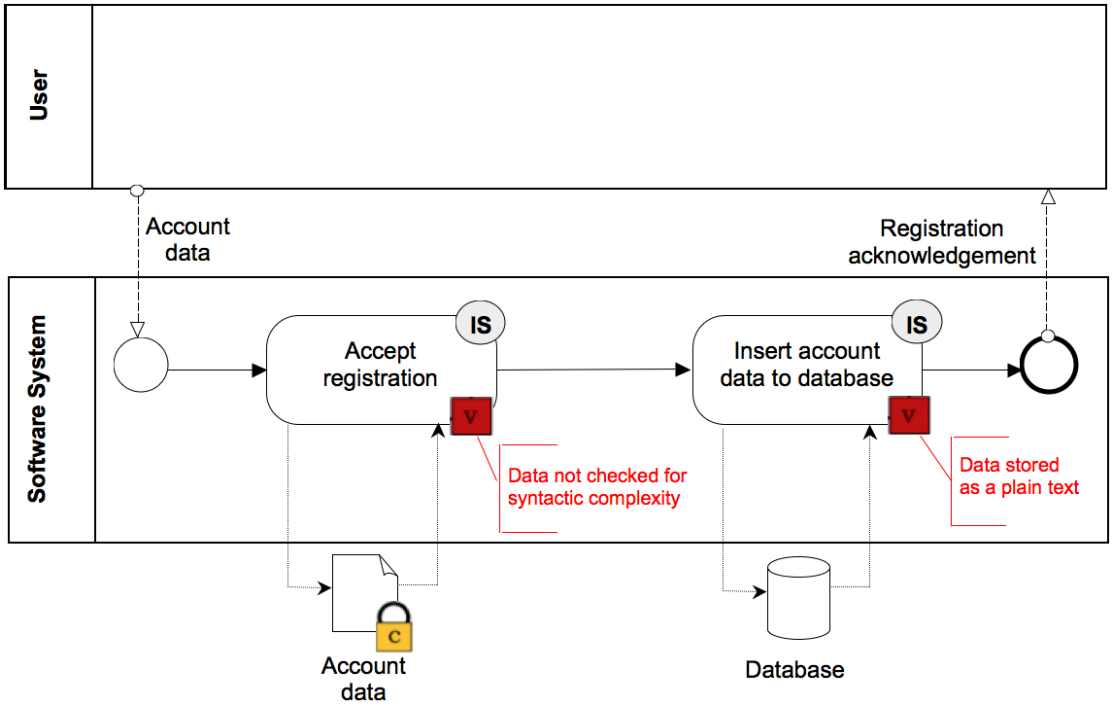
[12] R. Matulevicius, "MTAT.03.246 Principles of Secure Software Design", University of Tartu, https://courses.cs.ut.ee/2014/ssd/spring, April 2014.

Appendix

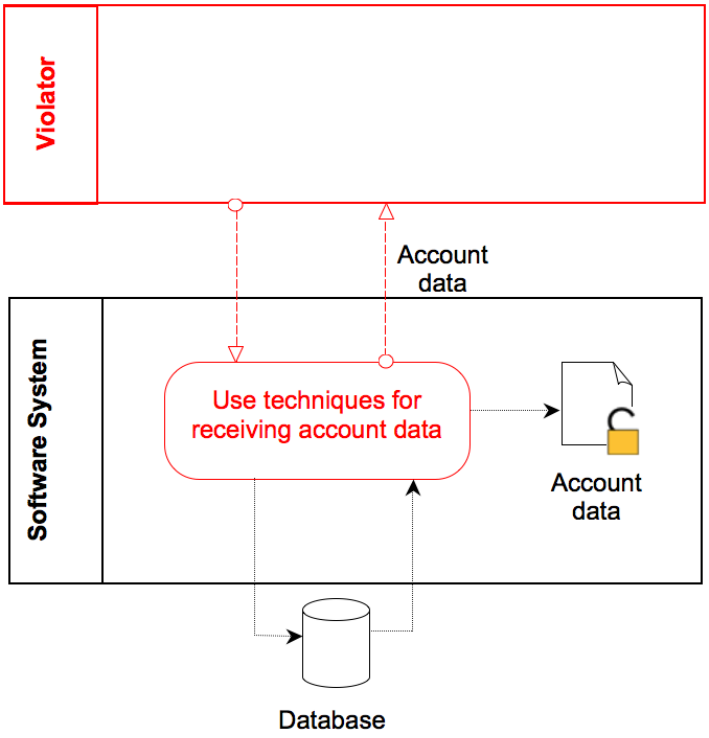
I. Attachements

		What is …? (ISSRM concepts)	Which language construct express …?
Asset	Business asset		
	IS asset		
Security criterion			
Risk			
Impact			
Event			
Vulnerability			
Threat			
Threat agent			
Attack method			
Risk treatment decision			
Security requirement			
Control			

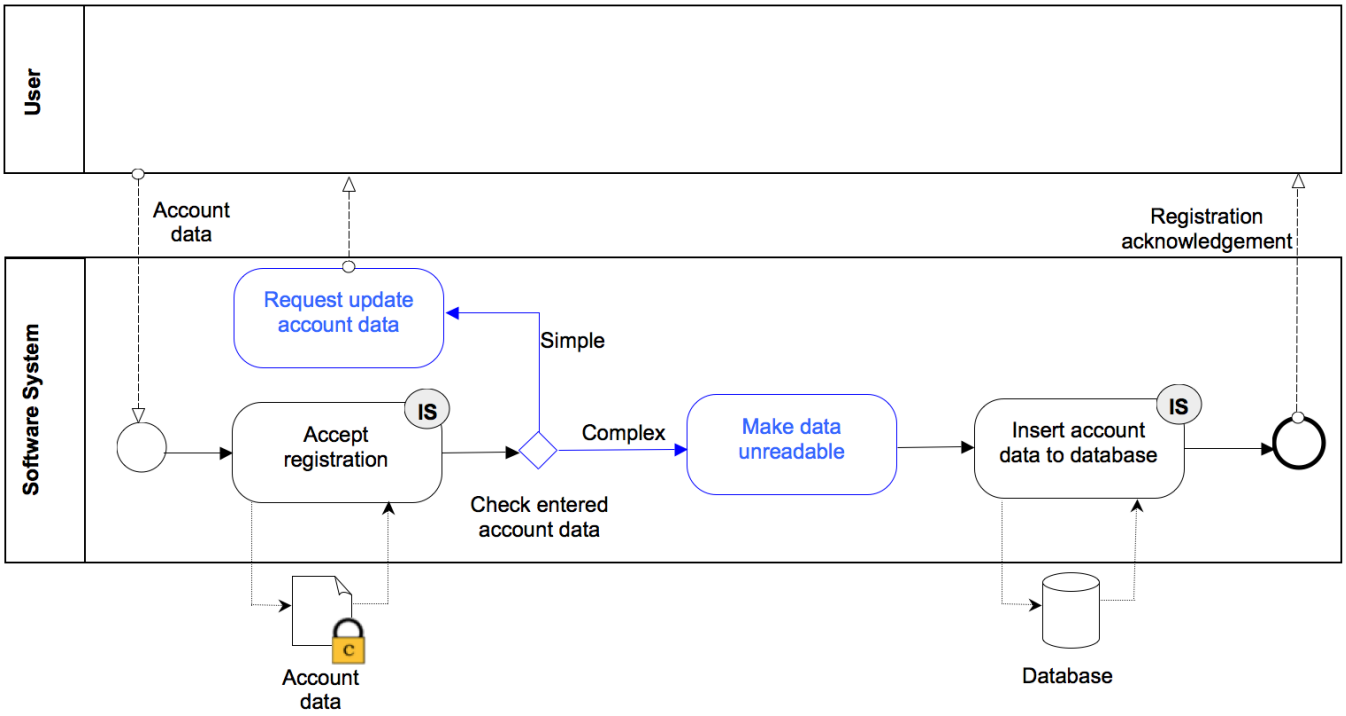
APPENDIX 1: SURVEY FORM FOR ALL OF THE PRESENTED MODELS



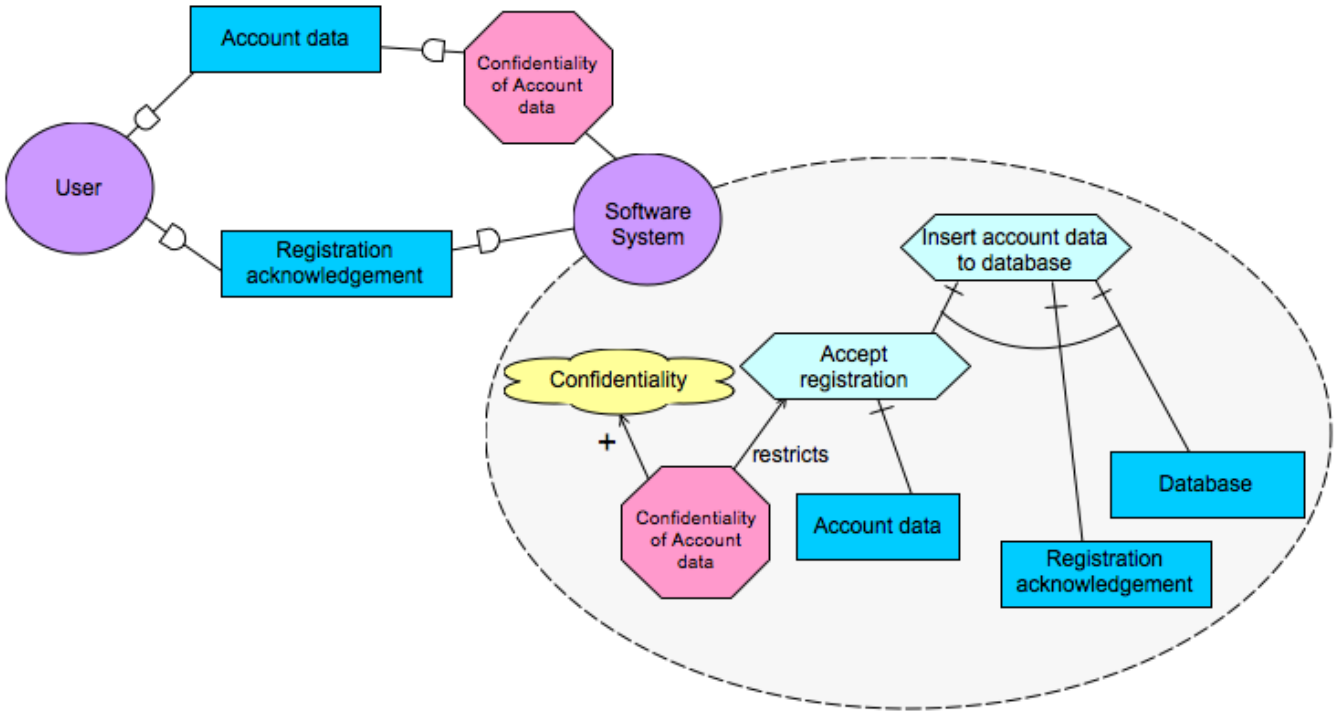
APPENDIX 2: BPMN - ASSET



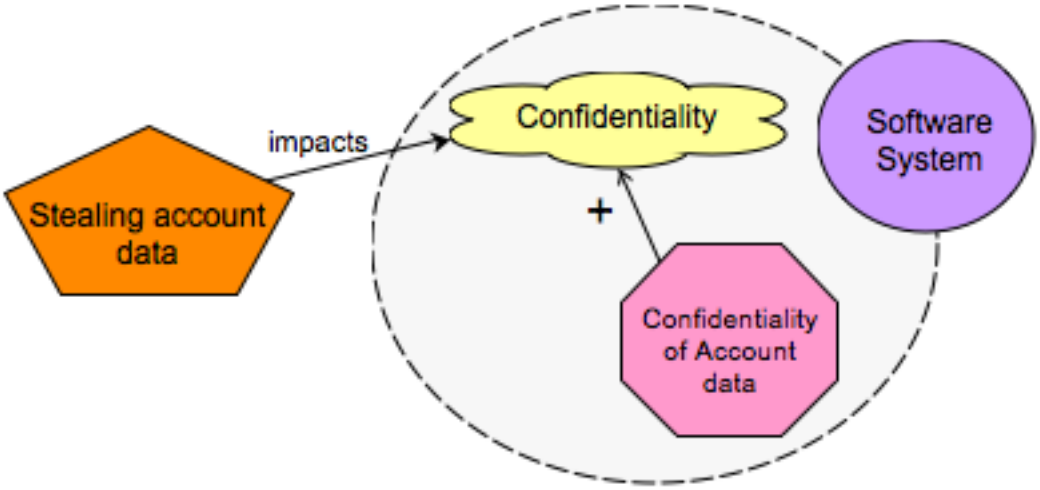
APPENDIX 3: BPMN – RISK



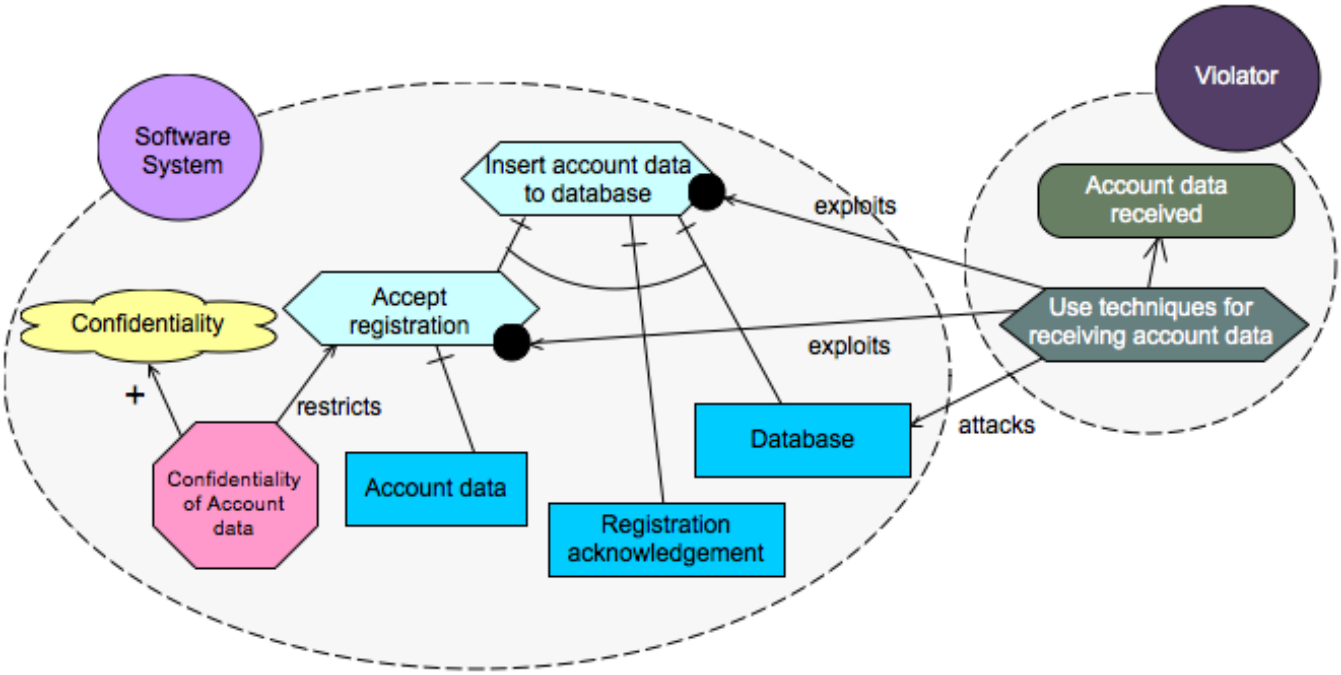
APPENDIX 4: BPMN - RISK-TREATMENT



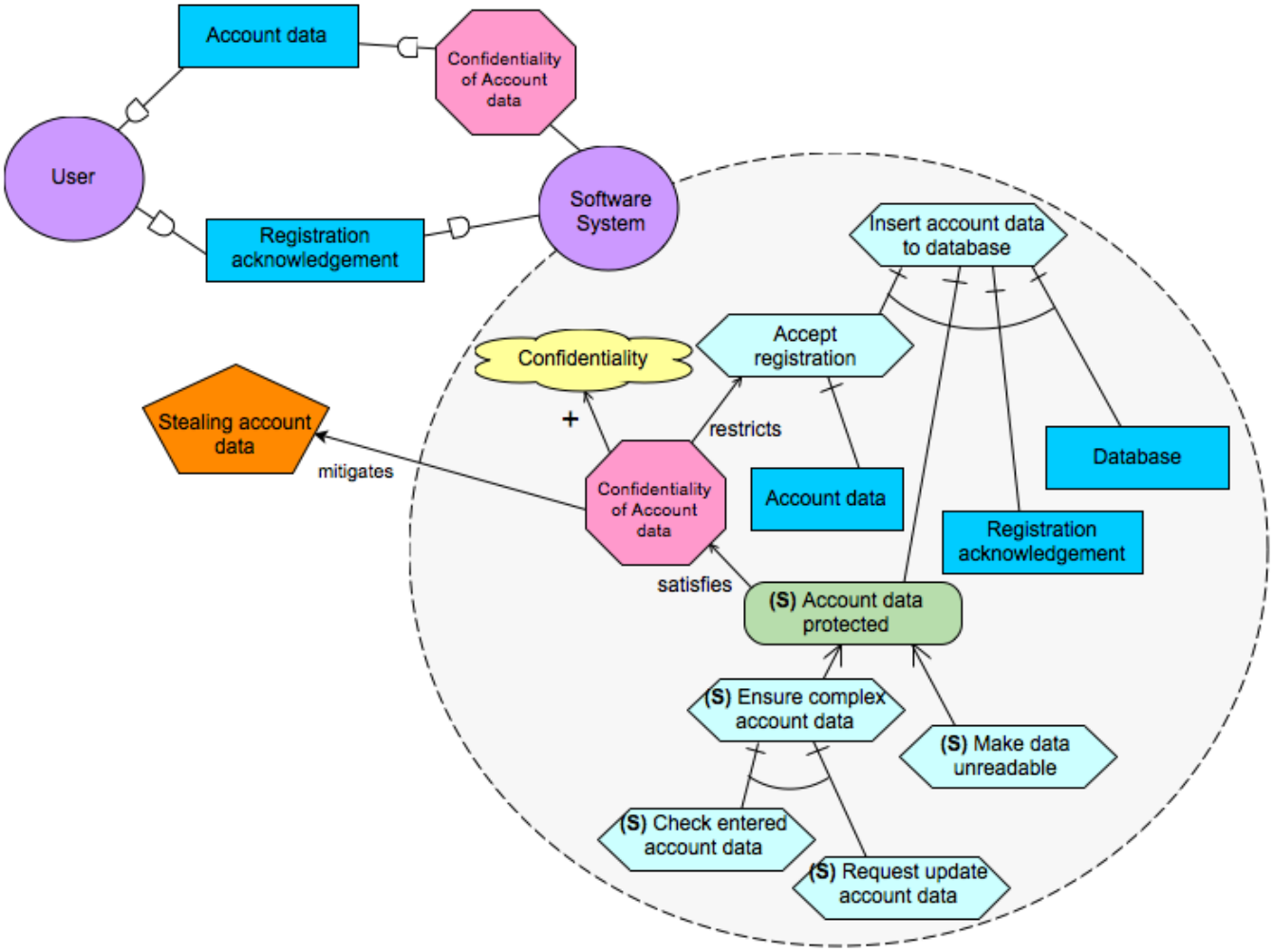
APPENDIX 5: SECURE TROPOS – ASSET



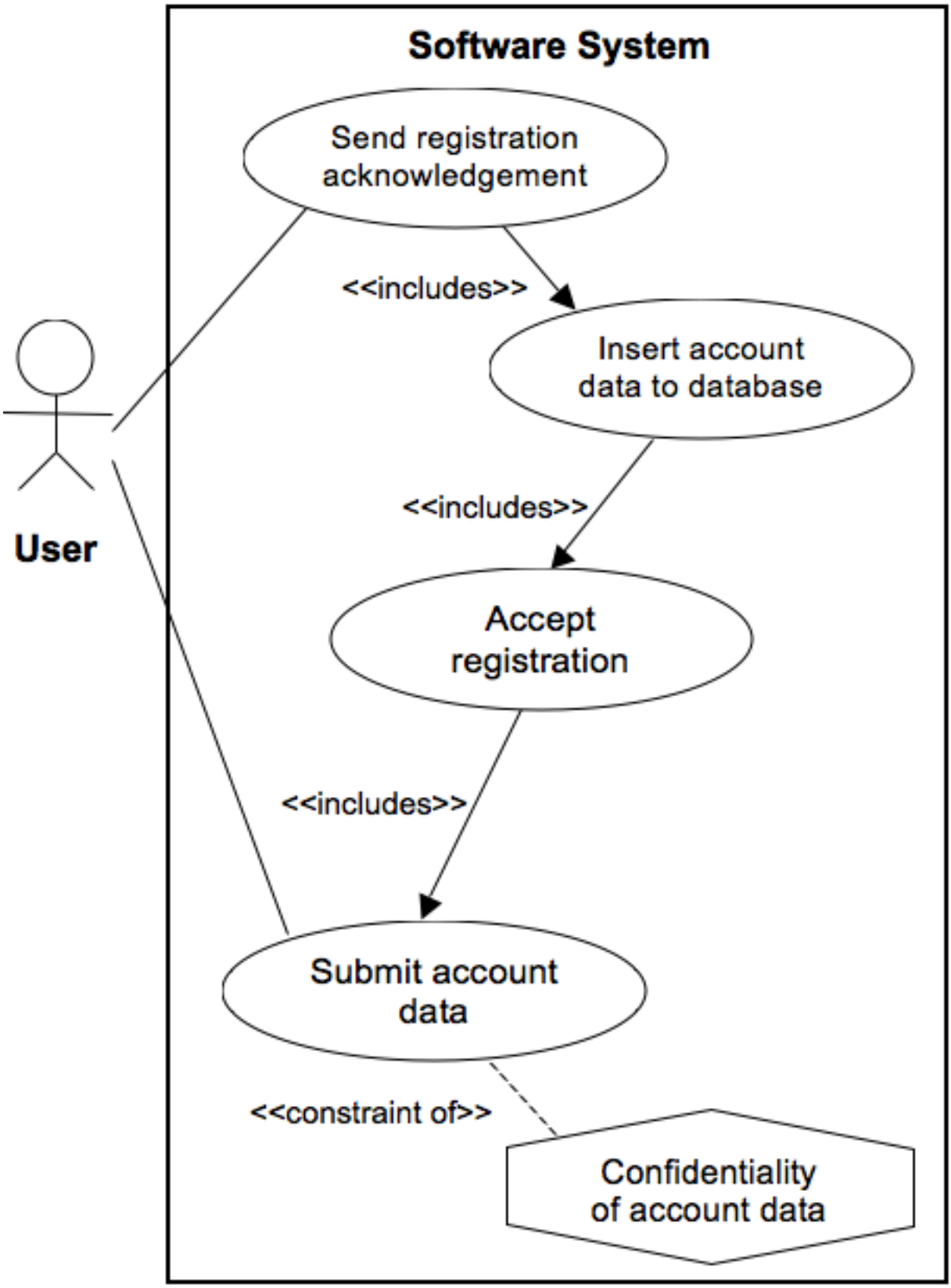
APPENDIX 6: SECURE TROPOS - RISK 1



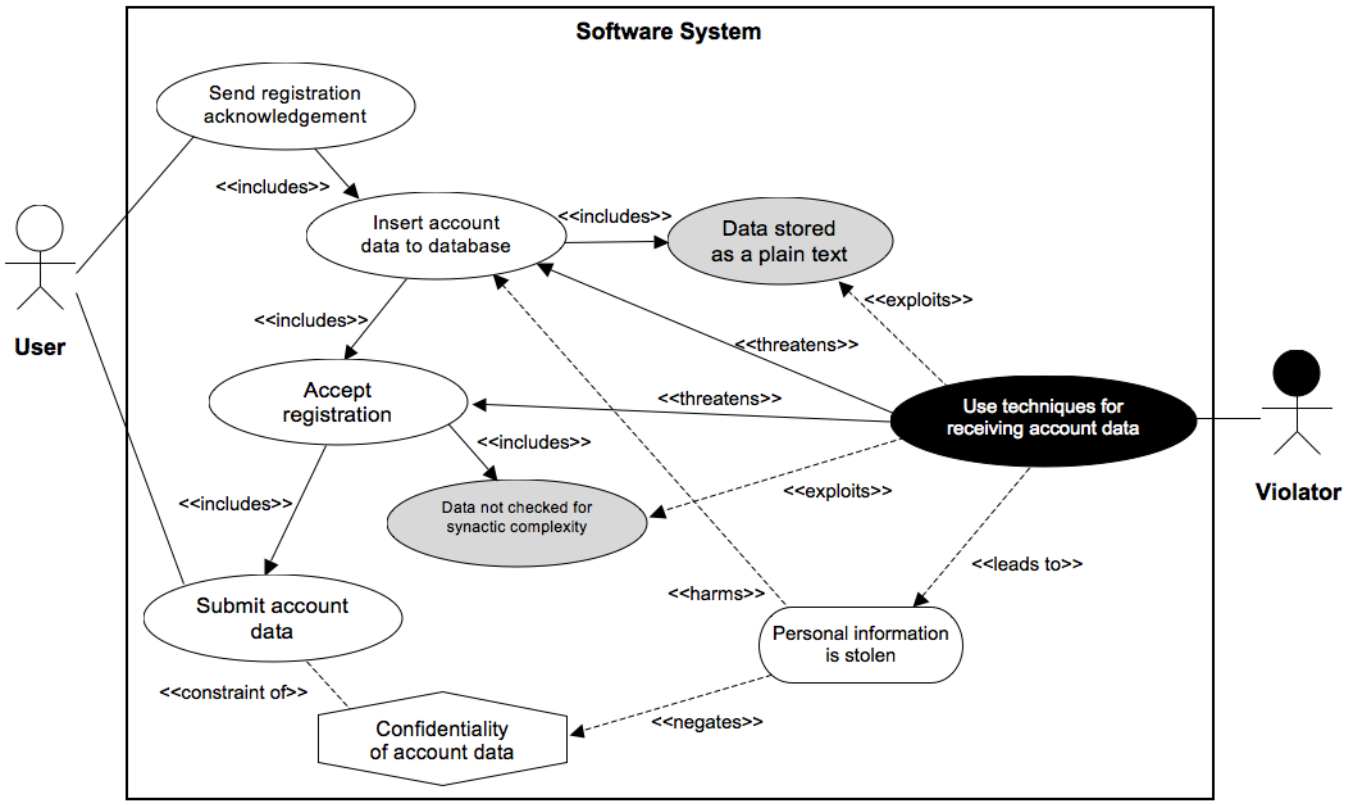
APPENDIX 7: SECURE TROPOS - RISK 2



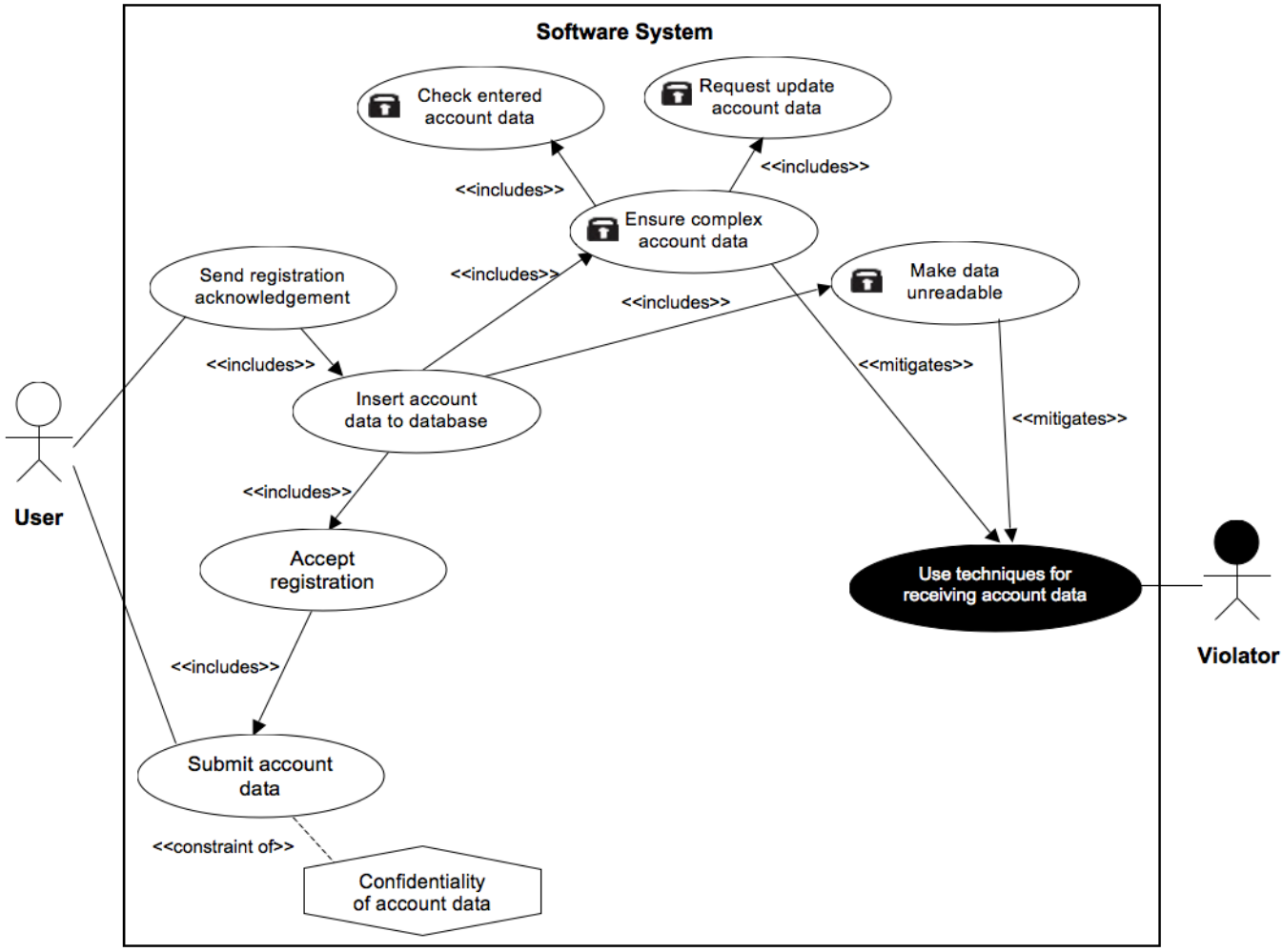
APPENDIX 8: SECURE TROPOS - RISK-TREATMENT



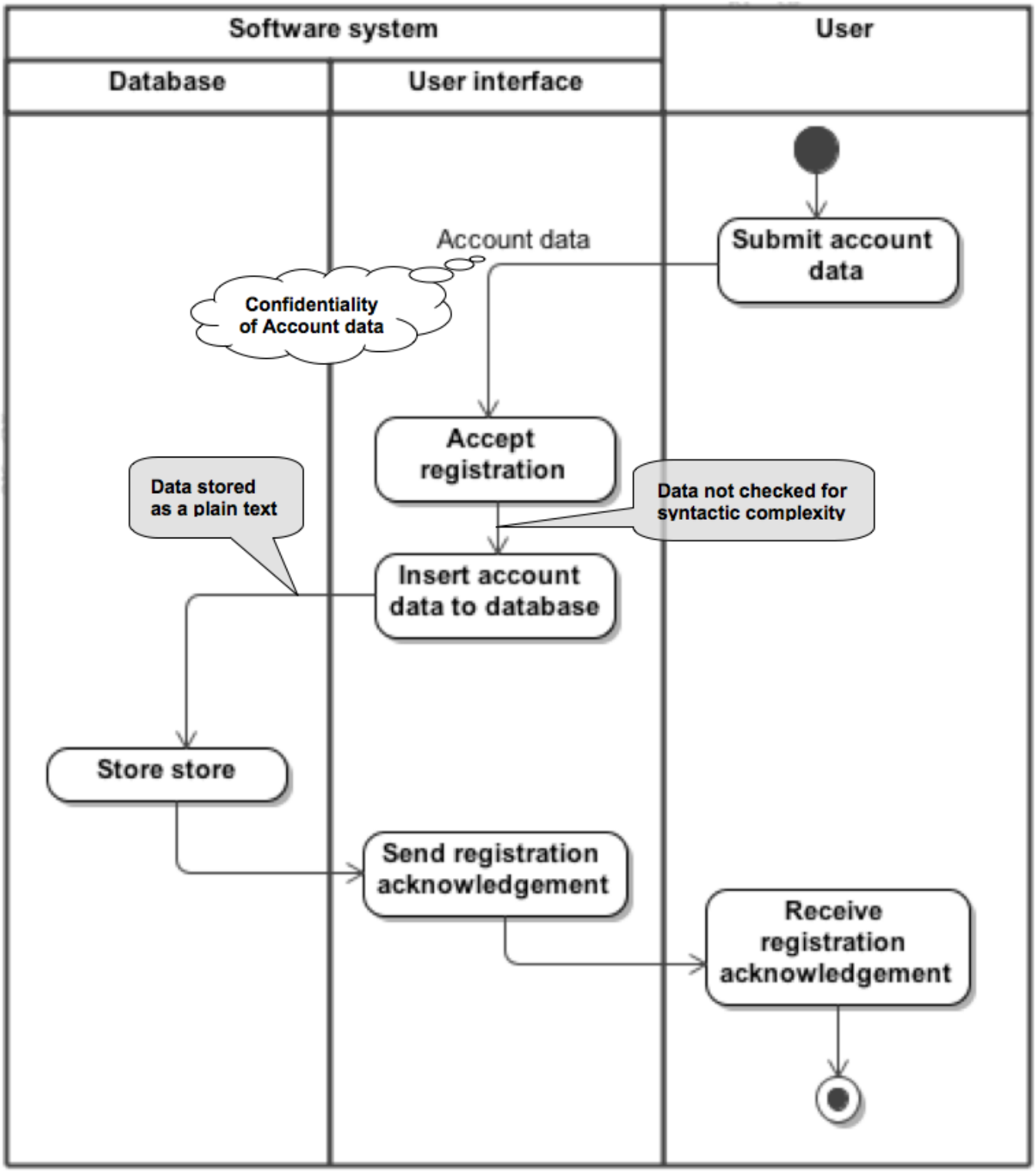
APPENDIX 9: MISUSE CASE – ASSET



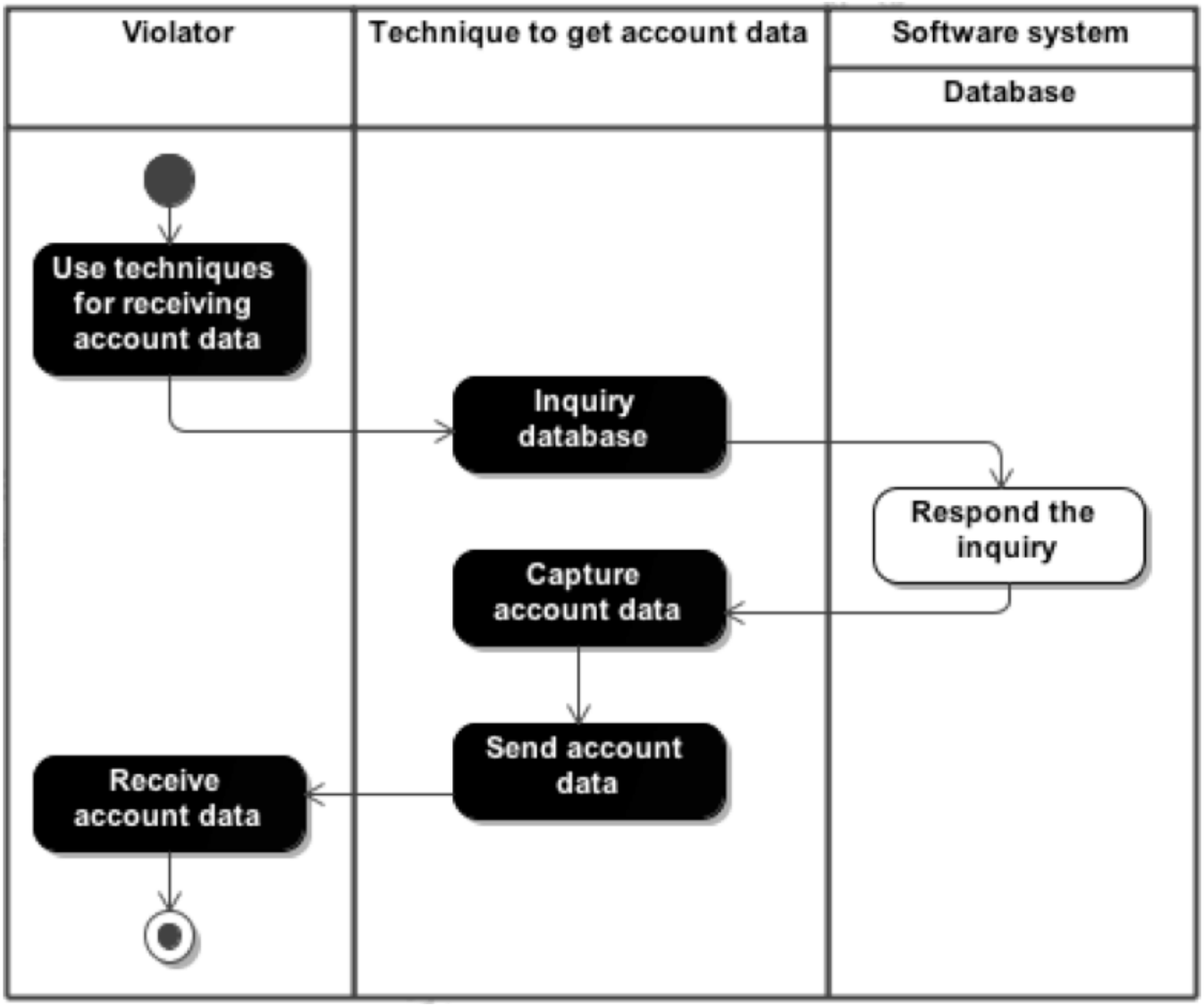
APPENDIX 10: MISUSE CASE – RISK



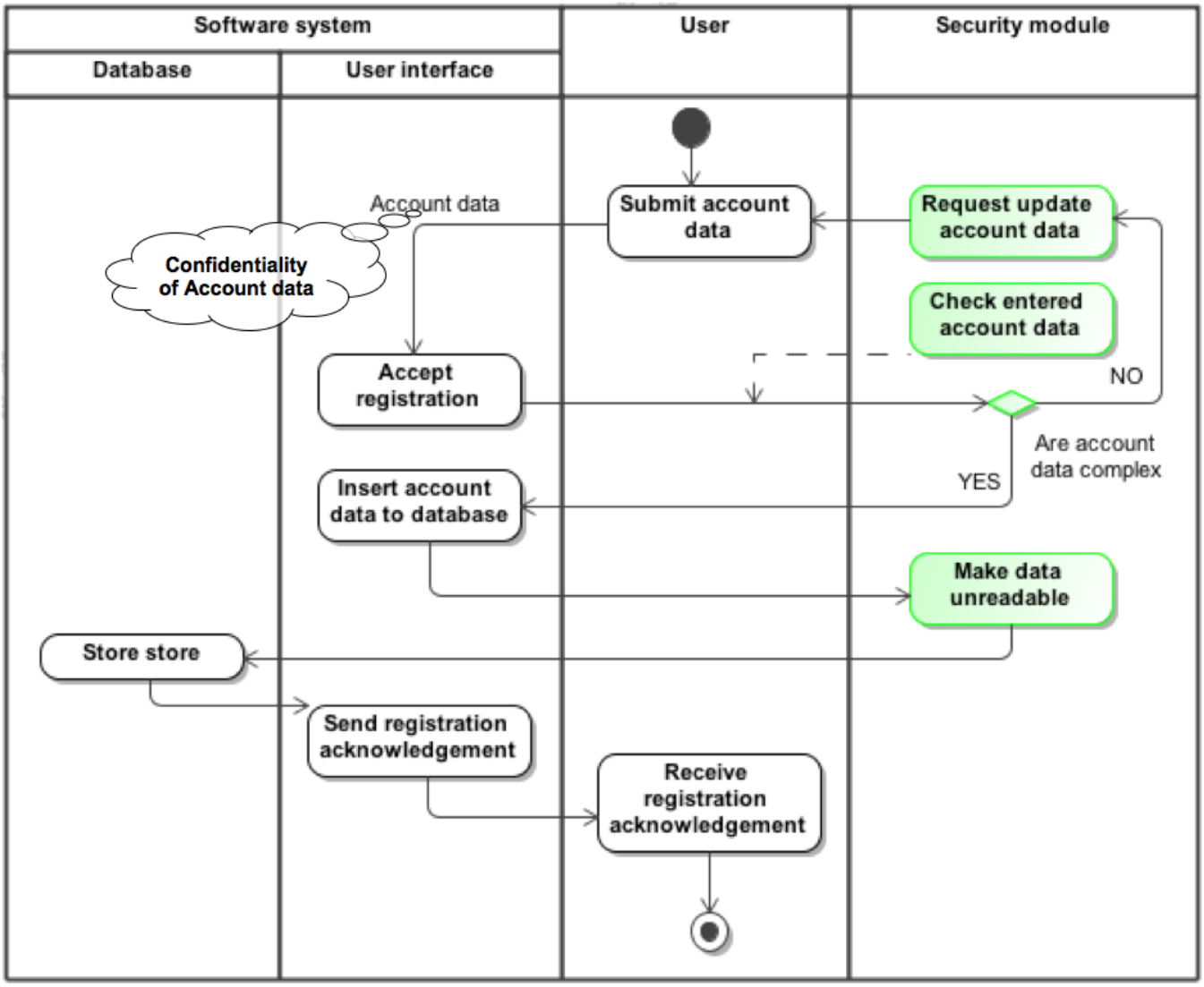
APPENDIX 11: MISUSE CASE - RISK-TREATMENT



APPENDIX 12: MAL-ACTIVITY DIAGRAM – ASSET



APPENDIX 13: MAL-ACTIVITY DIAGRAM – RISK



APPENDIX 14: MAL-ACTIVITY DIAGRAM - RISK-TREATMENT

ISSRM Concepts		Security risk-oriented				Overall
		BPMN model	Secure Tropos model	Misuse case diagrams	Mal-activity diagrams	
Number of valid responses		7	8	10	14	39
Asset concepts	Business asset	86%	100%	60%	93%	**85%**
	IS asset	43%	100%	50%	64%	**64%**
	Security criterion	100%	100%	90%	86%	**94%**
Risk concepts	Risk	43%	75%	40%	29%	**47%**
	Impact	100%	88%	70%	43%	**75%**
	Event	57%	75%	50%	57%	**60%**
	Vulnerability	71%	88%	70%	14%	**61%**
	Threat	71%	63%	50%	29%	**53%**
	Threat agent	100%	100%	90%	86%	**94%**
	Attack method	86%	63%	70%	57%	**69%**
Risk treatment concepts	Risk treatment	57%	50%	60%	50%	**54%**
	Security requirement	71%	50%	50%	43%	**54%**
	Control	0%	*0%*	0%	29%	**7%**
	Overall	**74%**	**79%**	**63%**	**54%**	**63%**

APPENDIX 15: COMPREHENSION OF THE MODELS 2013

ISSRM Concepts		Security risk-oriented				Overall
		BPMN model	**Secure Tropos model**	**Misuse case diagrams**	**Mal-activity diagrams**	
Number of valid responses		7	8	10	14	39
Asset concepts	Business asset	86%	38%	70%	7%	**46%**
	IS asset	29%	38%	40%	14%	**30%**
	Security criterion	43%	38%	50%	14%	**36%**
Risk concepts	Risk	57%	25%	50%	0%	**33%**
	Impact	43%	13%	40%	0%	**24%**
	Event	43%	13%	40%	0%	**19%**
	Vulnerability	29%	25%	20%	0%	**41%**
	Threat	57%	25%	50%	7%	**35%**
	Threat agent	100%	38%	60%	14%	**53%**
	Attack method	29%	25%	50%	7%	**28%**
Risk treatment concepts	Risk treatment	43%	13%	50%	43%	**37%**
	Security requirement	71%	25%	25%	7%	**32%**
	Control	43%	*0%*	60%	7%	**29%**
	Overall	**52%**	**24%**	**47%**	**9%**	**34%**

APPENDIX 16: LANGUAGE CONSTRUCTS 2013

II. License

Non-exclusive licence to reproduce thesis and make thesis public

I, **Andrei Proskurin** (date of birth: 23.07.1991),

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to:
   1.1. reproduce, for the purpose of preservation and making available to the public, including for addition to the DSpace digital archives until expiry of the term of validity of the copyright, and
   1.2. make available to the public via the web environment of the University of Tartu, including via the DSpace digital archives until expiry of the term of validity of the copyright,

of my thesis

A Comparison of Security Modelling Languages used for Security Risk Management,

supervised by Raimundas Matulevičius ,

2. I am aware of the fact that the author retains these rights.

3. I certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartu, **14.05.2014**