

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

Iuliia Tovstukha

**Management of Security Risks in the
Enterprise Architecture using ArchiMate and
Mal-activities**

Master's Thesis (30 EAP)

Supervisor(s): Raimundas Matulevičius

Tartu 2014

Management of Security Risks in the Enterprise Architecture using ArchiMate and Mal-activities

Abstract:

Security level of the enterprise is one of the main elements that should be taken under control in the organization. It is difficult to maintain high security level of Information System. Since development of enterprise architecture is targeted on continues business flow modeling, it sometimes does not take into account security requirements.

The paper provides an approach to improve security countermeasures to contribute with secure Enterprise Architecture. Filling the gap between Enterprise Architecture model and Security Risk Management is done through Information System Security Risk Management domain model (ISSRM). To build the Enterprise Architecture model, ArchiMate modelling language is being used. Among different risk-oriented languages, selection was done in favor of Mal-activity diagrams, which help to provide visual concept of Security Risk Management. Structured alignment can show the mapping between aforementioned terms and provide the information about most vulnerable points of the system. The maintenance of security level will help to make business flow independent from the state of Information System.

The outcome of this paper is an alignment tables and rules between ArchiMate and Mal-activity diagrams. The mapping link between these two languages is ISSRM. Validation of our approach is done on the example, which is taken from CoCoME case study. It is shown on number of illustrative pictures. After getting the results, there is a comparison of the output between presented method and approach developed by Grandry *et.al.* (2013).

Keywords:

Information System, Information System Security Risk Management, Enterprise Architecture, Enterprise Architecture model, security countermeasures, Security Risk Management, risk-oriented modelling languages, ArchiMate, Mal-activity diagrams.

Turvariskide juhtimine ettevõtte arhitektuuris kasutades tehnikaid ArchiMate ja Mal-activities

Lühikokkuvõte:

Turvalisuse tase on ettevõtte üks peamisi elemente, mida tuleb organisatsioonis kontrollida. Kui ettevõtte äri arengut modelleeritakse on eesmärgiks katkematu ettevõtlus, aga tihti ei võeta sellega arvesse turvanõudeid. Selliselt on aga infosüsteemi kõrget turvalisuse taset väga raske säilitada.

Selles dokumendis käsitletakse lähenemisviisi, mis parandab julgeoleku vastumeetmeid, et sellel läbi aidata ettevõtte arhitektuuri turvalisemaks muuta. Ettevõtte arhitektuurimudeli ja turvariski juhtimise vaheliste soeste leidmine toimub läbi Infosüsteemi turvariskide juhtimise domeeni mudeli (ISSRM). Ettevõtte arhitektuuri modelleerimiseks on kasutatud ArchiMate modelleerimiskeelt. Paljudest riskide kirjeldamise keeltest on sobilikum mal-activity (pahatahtlikute tegevuste) diagrammid, sest see aitab julgeoleku riskide juhtimist kõige paremini visualiseerida. Struktureeritud joondus aitab üldnimetatud keelte vahelisi seoseid näidata ning annab informatsiooni kõige haavatavamate punktide kohta süsteemis. Turvalisuse taseme säilitamine aitab ettevõttel äritegevust viia sõltumatuks infosüsteemist.

Selle dokumendi tulemuseks on ArchiMate ja Mal-activity diagrammide vahelised seostetabelid ja reeglid. Nende kahe keele vaheliseks seoseks on ISSRM. Kirjeldatud lähenemise valideerimine on läbi viidud ühe näite põhjal, mis on võetud CoCoME juhtumiuuringust. Näite põhjal on loodud mitmeid illustreerivaid pilte valideerimise kohta. Kõige viimasena on kirjeldatud meetodiga saadud tulemust võrreldud Grandy *et.al.* (2013) poolt arendatud lähenemisega.

Võtmesõnad:

Infosüsteem, Infosüsteemi turvariskide juhtimine, ettevõtte arhitektuur, ettevõtte arhitektuuri mudel, julgeoleku vastumeetmed, turvariskide juhtimine, riskidele orjenteeritud modelleerimiskeeled, ArchiMate, mal-activity diagrammid.

Table of Contents

Abstract.....	2
Lühikokkuvõte.....	3
Table of Contents	4
List of Figures.....	6
List of Tables.....	6
List of Abbreviations	7
1 Introduction	8
1.1 Research question and contribution	8
1.2 Scope.....	9
1.3 Structure.....	9
2 Security Risk Management	10
2.1 Methods and standards for Security Risk Management	10
2.2 DITSCAP Requirements Domain Model	11
2.3 ISSRM Domain Model	11
2.4 Comparison of Security Risk Management Domain Model.....	12
2.5 Summary.....	13
3 Security Risk-oriented Languages	14
3.1 Comparison of security risk-oriented modeling languages	14
3.2 Mal-activity diagrams	14
3.3 Summary.....	16
4 Enterprise Architecture Management.....	17
4.1 Enterprise Architecture Management Approaches	17
4.2 ArchiMate	18
4.3 Illustrated example.....	19
4.4 Mapping of ISSRM and ArchiMate.....	20
4.5 Summary.....	20
5 Alignment of Enterprise Architecture and Mal-activities	22
5.1 Method overview	22
5.2 Identification of Assets to Protect.....	23
5.2.1 ArchiMate Alignment to ISSRM: Asset Model.....	23
5.2.2 Asset Identification Example.....	24
5.3 Transformation to Mal activities.....	27
5.3.1 Transformation rules	27
5.3.2 Transformation example	28
5.4 Security Risk Management using Mal-activities	29
5.5 Transformation to ArchiMate	30
5.5.1 ArchiMate Alignment to ISSRM: Risk treatment-related concept.....	30
5.5.2 Transformation rules	35
5.5.3 Risk treatment in ArchiMate	36

5.6	Summary	37
6	Validation	39
6.1	Research question and method	39
6.2	Summary of results	40
6.3	Discussion	41
6.4	Threats of validity	41
6.5	Summary	42
7	Conclusions and future work	43
7.1	Limitations	43
7.2	Conclusions	43
7.2.1	The Institut Luxembourgeois de la Normalisation, de l'Accréditation	43
7.2.2	Answer to RQ	44
7.3	Future work	44
8	References	45
	Appendix	49
I.	Alignment of modeling languages with ISSRM DM	49
II.	License	53

List of Figures

Figure 2.1. DITSCAP Risk and Requirements DM adapted from [16]	11
Figure 2.2. ISSRM DM adapted from [38]	12
Figure 3.1. MAD presentation of ISSRM asset-related concept	15
Figure 3.2. MAD presentation of ISSRM risk-related concept.....	16
Figure 3.3. MAD presentation of ISSRM risk treatment-related concept.....	16
Figure 4.1. ArchiMate Framework adapted from [48]	18
Figure 4.2. EA model of Server Room example built with ArchiMate.....	19
Figure 4.3. EA model of Server Room example based on Grandry <i>et.al.</i> approach	21
Figure 5.1. Method algorithm diagram.....	22
Figure 5.2. EA model of Server room example defined for chosen business asset.....	27
Figure 5.3. MAD asset-related model for Server room example	29
Figure 5.4.a. MAD risk-related model for Server Room example	31
Figure 5.4.b. MAD risk-related model for Server Room example.....	32
Figure 5.5.a. MAD risk treatment-related model for Server Room example	33
Figure 5.5.b. MAD risk treatment-related model for Server Room example.....	34
Figure 5.6. EA model of Server Room example after method implementation	38
Figure 6.1. Validation steps	39

List of Tables

Table 2.1. DITSCAP Requirements DM and ISSRM DM comparison	13
Table 5.2. ArchiMate and ISSRM asset-related concept: general alignment.....	24
Table 5.1. ArchiMate and ISSRM asset-related concept alignment.....	25
Table 5.3. ArchiMate and ISSRM alignment based on Server Room example	16
Table 5.4. Alignment between ArchiMate and MAD: asset-related concept.....	28
Table 5.5. Alignment between ArchiMate and MAD based on Server room example.....	28
Table 5.6. ArchiMate and ISSRM risk treatment-related concept general alignment	35
Table 5.7. Alignment between ArchiMate and MAD for risk treatment-related concept...	36
Table 5.8. Alignment between MAD and ArchiMate based on Server Room example	36

List of Abbreviations

CC	Common Criteria
CORAS	Risk Assessment of Security Critical Systems
DITSCAP	Defense Information Technology Security Certification and Accreditation Process
DM	Domain model
EA	Enterprise Architecture
EAM	Enterprise Architecture Management
EBIOS	Expression des Besoins et Identification des Objectifs de Securite
GERAM	Generalized Enterprise Reference Architecture and Methodology
IEC	International Electrotechnical Commision
IS	Information System
ISO	International Organization for Standartization
IT	Information Technology
ISSRM	Information System Security Risk Management
MAD	Mal-activity diagram
MEHARI	Methode Harmonisee d'Analyse du Risque Informatique
NIST	National Institute of Standards and Technology
OCTAVE	Operationally Critical Threat, Asset, and vulnerability Evaluation
PIL	Product Identifiers List
RM	Risk Management
SRM	Security Risk Management

1 Introduction

Nowadays, the term of security is becoming more important and widespread. Information System (IS) is already integral part of every business. It supports business flow of organization and helps employees to operate with different business processes. Unfortunately, security concept is not the main point, which is taken into account during Enterprise Architecture (EA) development. The emphasis is made on the continuity of the business flow, but not on the maintenance of security level. However improvement of the security of the organization will positively influence on all business processes.

When EA is already developed and business is running, is it difficult to discover the vulnerabilities before the attacker will use them to violate the system. Possible risks could be mitigated through implementation of controls. Search of these countermeasures is made through risk analysis. Unfortunately after controls are defined for some particular asset, it is not visible how implementation of these countermeasures will influence on the whole EA. That's why the necessity of designing of new methodologies or investigating into extensions of existing approaches for Security Risk Management (SRM) and EA alignment remains actual and motivating.

1.1 Research question and contribution

The main research question of this paper is:

RQ: *How to align Enterprise Architecture and Security Risk Management?*

This paper targets to show how to use existed information from EA model in SRM. To make this happened firstly it is needed to go through SRM concepts and methods and chose the one that is the most suitable for us. The exploration of them will help to understand how security risks are managed. After the SRM Domain Model (DM) is chosen, it is necessary to find the modeling language that supports defined DM. Analyzing and comparing different modeling languages, Mal-activity diagrams (MAD) [44] were chosen as the language for further contribution. The alignment between chosen SRM DM (Information System Security Risk Management (ISSRM) [12]) and MAD was already made by Chowdhury *et. al.* (2012) [8]. To answer the defined research question, it is also necessary to find the modeling language to present EA model. Through analysis of different frameworks and methods, ArchiMate was chosen as a modeling language for EA model development. Despite there is already an alignment between the ArchiMate and ISSRM [18], it was decided to made this alignment by ourselves, as it will be used for further mapping between ArchiMate and MAD. Unlike the already defined alignment (presented by Grandry *et.al.* (2013) [18]) our purpose is to map ArchiMate and ISSRM without using any additional models from both sides. The research results are gathered into transformation tables between ArchiMate and MAD.

It is not enough to make transformation just from EA model into SRM model. Since even if countermeasures are defined correctly, there is no visible influence on model of enterprise IS in general. That's why it is important to make the transformation back from SRM model to EA model. Although it is done through the same modeling languages, it has more analysts work than previous transformation. It happens, since in second transformation security countermeasures and controls could be presented through many elements of ArchiMate [48]. That's why there are some general rules defined (even more as guideline) and analyst should think carefully how and where to transfer the required controls.

1.2 Scope

SRM could be done in different ways through usage of different concepts, method and standards. This work is specified on usage of ISSRM [12], which helps to define the same terminology for two modeling languages (ArchiMate [48] and MAD [44]), which are used in this work. EA could be presented through different frameworks and modeling languages. ArchiMate was chosen for this work among proposed variety. MAD is being used to present ISSRM though different concepts and map it to the EA representation.

1.3 Structure

The thesis is structured in eight chapters, which are conditionally organizing 3 big parts. The first part is background. It is presented by three chapters, which give an overview of SRM, Security risk-oriented languages and Enterprise Architecture Management (EAM). Each of these chapters contains an overview of existing concepts/languages/frameworks and is provided with examples and explanations. The discussion in Chapter 2 is made to justify the choice of ISSRM in our research. Chapter 3 is dedicated to Security risk-oriented languages and provided with an example of usage of the chosen one. Chapter 4 shows how ArchiMate could be applied within the example taken from study case. It also contains a description and example-based illustration of the alignment of ArchiMate to ISSRM taken from Grandry *et.al.* (2013) approach.

The second part presents a contribution of the proposed method. It is presented within one chapter. Chapter 5 starts with method overview. Each next sub-chapter corresponds to the blocks from Figure 5.1, so each step of our method is described in separate sub-chapter. Chapter 5 also has an illustration on application of the proposed approach.

The third part is validation, which describes the comparison of approach presented in this work and Grandry *et al.* (2013) concept. The comparison is made according to the defined criteria. All this is described in Chapter 6, which also contains information about threats of validity and summary results. Last but not least chapter of validation part is Chapter 7, which presents the summary, conclusions and limitations of the whole work. It also contains the future perspectives for work improvement.

2 Security Risk Management

The Risk Management (RM) is a set of coordinated activities, the main goal of which is to control an organization with respect to the possibly occurred risks. Methods and standards for identification threats, vulnerabilities and risks could be divided into 4 categories: RM standards, security standards, Security Risk Management (SRM) standards, SRM methods. AS/NZS 4360 [3], Common criteria (CC) [10], EBIOS [13], MEHARI [9] *etc.* could be aligned as examples. Mostly all of them consist of process guidelines that help to identify vulnerable assets, determine security objective, and assess risks as well as define and implement security requirements to treat the risk [12]. From all this variety we will stop our attention on two of them: Defense Information Technology Security Certification and Accreditation Process DITSCAP [32] and ISSRM. The motivation of reason, why ISSRM was chosen as methodology for current research, is also presented in this chapter.

2.1 Methods and standards for Security Risk Management

All of methods and standards for SRM have their advantages and disadvantages. For example, although the RM standards provide general considerations about RM, they are not so much security directed, which is not suitable in our case. To this category belong AS/NZS 4360 [3] and ISO/IEC Guide 73 [28].

In security standards category documents usually have security-specific terminology and sometimes some RM concepts, but they are not specifically focused on RM activities. These documents are ISO/IEC 13335 [25] and CC. CC is not acceptable for our research, because it is not completely aligned with IS security that is needed in our research. ISO/IEC 13335 is too much security specified and not as much RM specified. ISO/IEC 27001 [26], NIST 800-27 [15] and German BSI [21] are SRM standards. These standards are focused on RM activates through perspective of security. They provide prioritization, evaluation and implementation for the controls coming from the risk assessment process. The widest category is called SRM methods. Under this category we can separate such methods as EBIOS [13], MEHARI [9], OCTAVE [1], CRAMM [24] and CORAS [50]. One of the weaknesses of methods is lack of interoperability between these approaches and lack of alignment with standards. Although all of them consist of almost same steps (identification of the assets, threats, vulnerabilities, risk assessment, determination of security requirements), these methods cannot provide finished model as an outcome (besides CORAS method). The drawback of CORAS is disconnection from standard terminology [38]. The main disadvantage of mainly all aforementioned methods and standards is the way of output of the documents. It is composed in informal way, what is leading to the inconvenience in automatization. To sum up all limitations, we can consider that none of these methods is suitable for us.

According to defined goal in Chapter 1 we need to find an alignment between two different concepts. For better graphical understanding it should be done in the way of model. Hence, we will compare two concepts which could provide visible outcome. One of them is Defense Information Technology Security Certification and Accreditation Process (DITSCAP), which presents DITSCAP Requirements DM [17], and the second is ISSRM with ISSRM DM.

2.2 DITSCAP Requirements Domain Model

DITSCAP Requirements DM is used for effective decision-making activities regarding their interpretation, applicability, and implementation effectiveness in the IS [16]. Building of the model consists of different steps. In the center of the whole analysis stays the Certification and Accreditation (C&A) of the requirements. Security requirements based on C&A are defined in many regulatory documents, which could be even interconnected. Unfortunately these documents could have a different level of abstraction. To fulfill the main goal and support an overall risk-based strategy it is necessary to build DM. The Risk and Requirements (R&R) DM should consist of relevant risk components, such as threats and vulnerabilities of the assets to be protected and countermeasures to mitigate or reduce the vulnerabilities. The natural language description of basic risk components is taken from CC security model. They are extended and presented in the R&R DM, which is shown on the Figure 2.1.

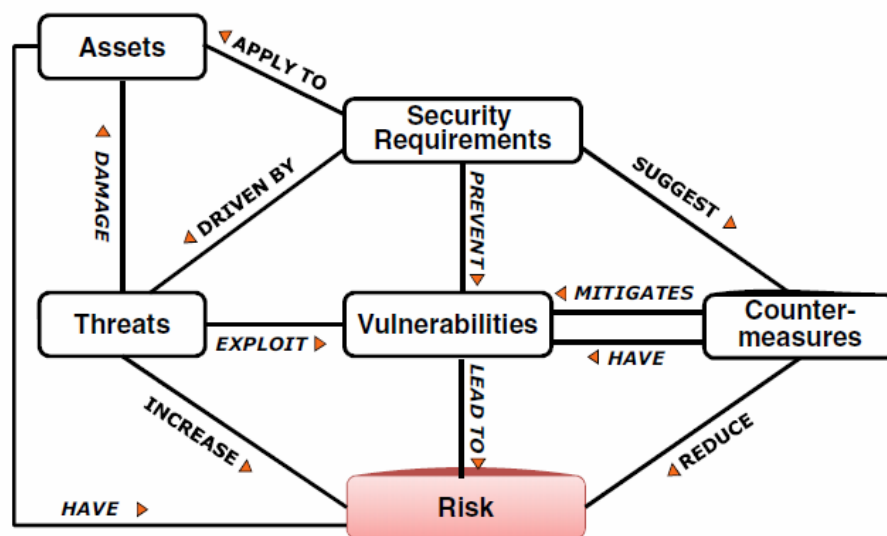


Figure 2.1. DITSCAP Risk and Requirements DM adapted from [16]

2.3 ISSRM Domain Model

ISSRM DM is the method, the main objective of which is defined as the protection of essential IS constituents against all harm to information security. ISSRM DM is structured around three groups of concepts: asset-related concepts, risk-related concepts and risk treatment-related concepts [38]. ISSRM DM (see Figure 2.2) supports definition of security for the main parts of information systems and addresses the IS security risk management process at its three different aforementioned conceptual levels [12].

ISSRM DM consists of 3 concepts: asset-related concepts, risk-related concepts and risk treatment-related concepts. The outcome of each gives us fundamental understanding about assets, vulnerabilities and threats. For example, *asset-related concepts* provide the information about assets of the system, which must be protected. The term asset in general stands for anything that has been valued for the organization and it is necessary for achieving its goals. In DM, which is presented in Figure 2.2, assets are divided into business assets and IS assets. IS assets supports business assets, which are aiming to achieve goals of organization. Furthermore criteria to guarantee asset security are described in these concepts. Security criterion identify which security criterion should be

ISSRM DM gives more information than DITSCAP DM, which shows the biggest advantage of this DM. For example, DITSCAP DM does not cover the risk treatment concept at all. It does not provide the information about possible controls that could be implemented in order to mitigate the risk.

The way of modeling the chosen DM is important in our research. In case of DITSCAP Requirements DM, the modeling phase will be done by GENeric Object Model meta-language. It is only one modeling language with which there is alignment. However this meta-language has it's toolkit with helps to automate it. This gives advantage for this DM. A number of modeling languages (like Mal-activities, Secure Tropos, Misuse Cases *etc.*) are aligned to the ISSRM DM. All of these languages have different purposes. The variety of them makes ISSRM DM more suitable for our research than DITSCAP DM.

Both models have their advantages and disadvantages. The usage of each of them could be more sufficient regarding to the situation. We defined the necessary criteria for comparison of two DMs, which could help to choice the most suitable model for the particular situation. The actual comparison is in the Table 2.1, where + means fully covered, - -not covered at all, +/- - not full covered.

Table 2.1. DITSCAP Requirements DM and ISSRM DM comparison

	Visualizati on as domain model	Alignment with modeling languages	Asset- related concept coverage	Risk- related concept coverage	Risk treatment- related concept coverage	Visible impact of implemented requirements
DITSCAP DM	+	+	+/-	+	-	-
ISSRM DM	+	+	+	+	+	+

2.5 Summary

We select the ISSRM DM for further contribution as it gives whole information for all concepts, when DITSCAP DM does not give any information about possible risk treatment and no differentiation between IS and business assets. ISSRM DM is the most suitable for our research as it could be extended with the help different security risk-oriented modeling languages. Moreover implementation of the suggested controls regarding to the chosen requirements could visibly prove the mitigation or reduction of vulnerability of the asset. Although DITSCAP Requirements DM is requirements directed, it does not give the full definition of assets, threats and vulnerabilities, which makes ISSRM DM more suitable for our research.

3 Security Risk-oriented Languages

Now there exist many security risk-oriented modeling languages, such as Secure Tropos [40], KAOS extension to security [30], BPMN extension to security risk management [42], UMLsec [29], SecureUML [33], Misuse cases[43], Mal-activity diagrams [44] *etc.* We will stop our attention on four modeling languages: Misuse cases, Mal-activity diagrams, BPMN, Secure Tropos. All these languages were previously aligned to ISSRM DM, what is suitable for us according to the Chapter 2.

3.1 Comparison of security risk-oriented modeling languages

Misuse cases [43] are an extension of use cases, in a way to detail common attempts to abuse the system. The misuse case diagram should be design for each malicious actor in order to show all possible abuses. The main goal of misuse cases is to describe the behavior that should not be allowed in the system [45]. The misuse case diagram extends use case diagram with 2 entities: misuse case and misuser. Misuse case is a sequence of actions that could be done by any person or software in order to harm the system. Misuser is the actor, who initiates the attack (misuse case).

Mal-activity diagram (MAD) [44] is designed to show a harmful behavior of security attackers on the IS. Firstly, in the mal-activity diagram a normal process is built, and then it is added with a set of malicious behavior. Inappropriate behavior is shown through mal-activities, mal-swimlane and mal-decision construct.

Business Process Model and Notation (BPMN) [42] is used for graphical representation of business processes flow in IS system. It shows specific business processes in a Business Process Diagram. The main goal of BPMN is specifying the gap between the business process design and implementation. The BPMN application is divided into three usage level: analytical modeling, executable modeling, and descriptive modeling.

Secure Tropos [40] supports modeling through 4 phases: early requirements analysis, late requirements analysis, architectural design and detailed design. It is based on iterative process: diagrams built on one phase are used to create diagrams on next phase. The whole process of modeling starts with identifying actors and list of goals for each actor. Then dependencies between the actors are defined, together with dependencies between actors and system.

All four modeling languages have an alignment with the ISSRM DM [40, 42, 43, 44]. Detailed alignment is presented in the Appendix I. Since they have different syntax, they could be used in different situations. MAD will be taken for further consideration, as it gives the full picture of required IS. This modeling language specifies the malicious actor and his potential activities against the system. The final model gives step-by-step guide of the system against attacker actions.

3.2 Mal-activity diagrams

The MAD will be presented through one example based on CoCoME study case [19]. One risk will be taken under consideration and observed through 3 steps of the ISSRM process. This example shows the correspondence between the employees and server room, and the way of how unauthorized person could potentially harm the correspondence. The risk is giving *unauthorized access into the server room*. In another words it shows how an

entrusted employee gets an unauthorized access to the server room, because of absence or lack of access privileges, and messes up the product identifiers in a database, which leads to the loss of integrity of the product identifier list (PIL) and constrains the correct selling process for the whole store. The impact of the risk could harm the PIL; the server room is not reliable, since anyone can access it. The integrity of a PIL will be negated. Furthermore this risk leads to stop the operation of a whole store and loss of customer trust and loyalty. The vulnerability of the IS is the lack or absence of access privileges to server room. The risk could be mitigated through - implementation of access control – magnetic cards, doors with PIN codes, implementation of RBAC; and monitoring the entrance of the server room.

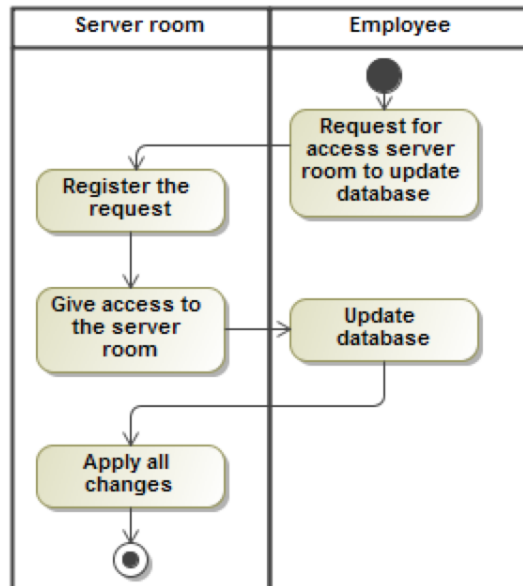


Figure 3.1. MAD presentation of ISSRM asset-related concept

Asset-related concept is presented in Figure 3.1. It is described through two swimlanes: *Employee* and *Server room*. In the *Employee* swimlane the business asset (*database*) is defined. *Server room* swimlane shows the constructs that are needed to support execution of the workflow. There is no construct for security criterion, but from the definition of the assets it is understandable how they could be negated.

In risk-related concept an *Attacker* presents the malicious actor and is defined through mal-swimlane (see Figure 3.2). The attack methods are defined through mal-swimlanes (*Social engineering* and *Hacker's computer*) and processes under this mal-swimlane (*Request for access to server room* and *Change data in database*). As an impact *Refer to boss' order*, *Connection of hacker's computer to the server* and *Getting database credentials* are defined. Unfortunately, the vulnerabilities are not presented in MAD as special element.

In risk treatment-related concept, which is presented in Figure 3.3, countermeasures for the system are defined. The separate *Security module* swimlane is created, where all possible controls are mentioned. Security requirements are defined as *Verification of identity* and *Checking access rights for identity*.

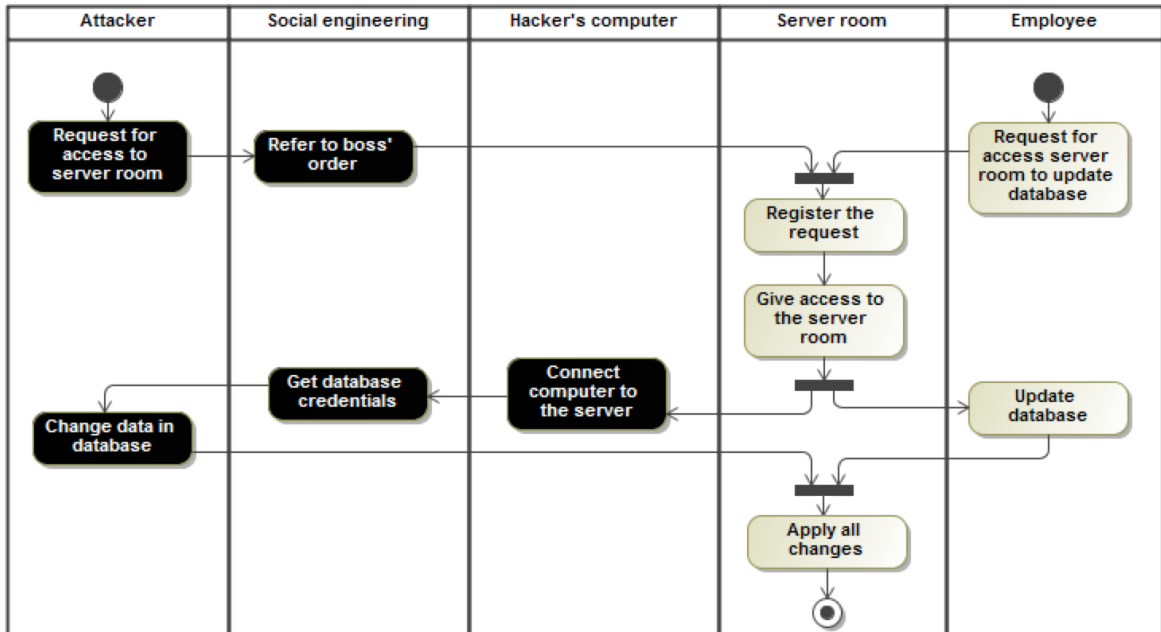


Figure 3.2. MAD presentation of ISSRM risk-related concept

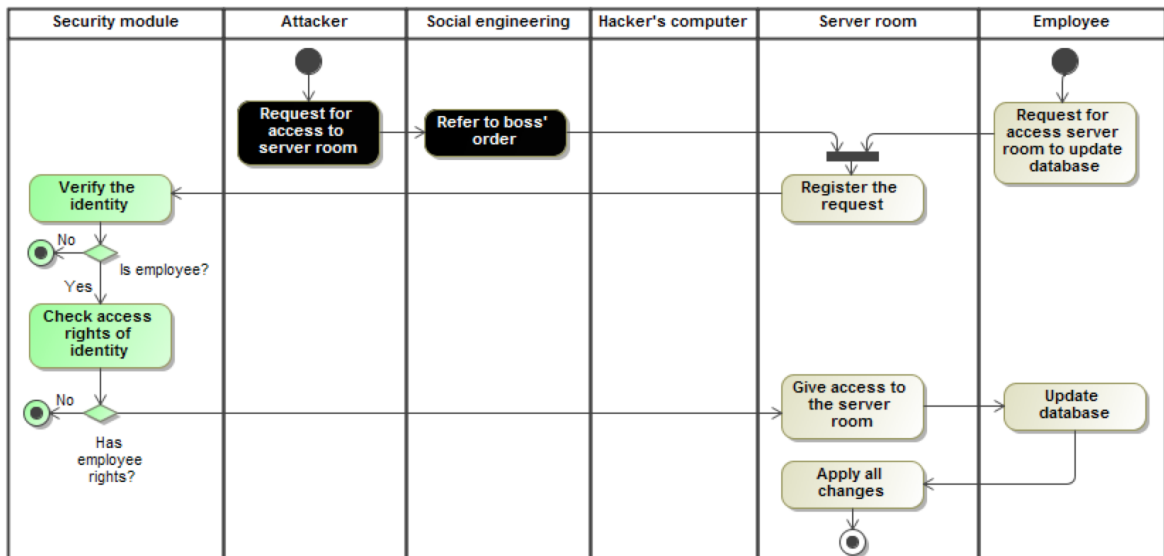


Figure 3.3. MAD presentation of ISSRM risk treatment-related concept

3.3 Summary

In this chapter different modeling languages for SRM were reviewed (Misuse cases, MAD, Secure Tropos and BPMN). MAD was presented in more details and shown through example based on CoCoME study case. Moreover MAD was chosen for further contribution of this work, since among mentioned languages it has the biggest emphasis on the attack process and attacker behavior. In other words it helps to add malicious activity into normal work process. One more advantage to choose MAD as modeling language is that it has more smooth and continuous move between requirement engineering and design stage.

4 Enterprise Architecture Management

There are different enterprise architecture frameworks, which show principles and practices for creation and usage of EA. The description of architecture consists of domains, layers or views. The model itself could be presented as matrix or diagram. To build the diagram, modeling language is needed, so this chapter is dedicated to different approaches to present EA.

4.1 Enterprise Architecture Management Approaches

Zachman framework [51] is EA framework, with the help of which formal and highly structured way of viewing and defining of an enterprise could be reached. It consists of a two-dimensional classification matrix. It is based on the intersection of six communication questions, which are What, Where, When, Why, Who and How, with five levels of reification, successively transforming the most abstract ideas into more concrete ideas. The basic idea behind the Zachman Framework is that the same complex thing or item can be described for different purposes in different ways using different types of descriptions (e.g., textual, graphical). This framework gives the 36 necessary categories for completely describing anything. The framework provides six different transformations of an abstract idea (not increasing in detail, but transforming) from six different perspectives.

An important drawback is the large number of cells, which is an obstacle for the practical applicability of the framework. Also, the relations between the different cells are not that well specified. Notwithstanding these drawbacks, Zachman is to be credited with providing the first comprehensive framework for EA, and his work is still widely used.

Generalized Enterprise Reference Architecture and Methodology (GERAM) [20] identifies the set of components recommended for usage in enterprise engineering. GERAM is an enterprise-reference architecture that models the whole life history of an enterprise integration project from its initial concept through its definition, functional design or specification, detailed design, physical implementation or construction, and finally operation to obsolescence.

The model proposed by GERAM has three dimensions: the life cycle dimension, the instantiation dimension allowing for different levels of controlled particularization, and the view dimension with four views: Entity Model Content view, Entity Purpose view, Entity Implementation view, and Entity Physical Manifestation view. Each view is further refined and might have a number of components.

Enterprise Architecture Meta-model [23] is divided in four main layers focusing on different levels of abstraction: business, the application layer, the technical layer and the physical layer. The different layers are interconnected by the associations of the meta-model that crosses the layer boundaries. Furthermore it is possible to provide the various stake-holders with different views on the enterprise architecture that show only specific types of artifacts.

ArchiMate [48] is one of the opened and independent enterprise EA modeling languages, which, with the help of business domains, supports the description, analysis and visualization of architecture. ArchiMate models follow a certain structure that is explained by means of an ‘analysis meta-model’. ArchiMate offers a common language for describing the construction and operation of business processes, organizational structures,

informational flows, IT systems, and technical infrastructure. An architecture framework is used to structure the concepts and relationships of the ArchiMate language. One of the objectives of the ArchiMate language is to define the relationships between concepts in different architecture domains. In ArchiMate there is three-layered view: the business, application and technology layers. Each layer is self-contained despite being a component of the integrated model, and caters to one or more architecture domains.

4.2 ArchiMate

The main goal of ArchiMate is to make a connection between the business and IT systems within one enterprise. ArchiMate is an approach, which visualizes the different architecture domains and shows their relations and dependencies. It also provides structure in representation of layers of the system. ArchiMate brings the visual presentation of the system, which is easily could be brought through the time. ArchiMate could be presented through 2 viewpoints, which define structure of ArchiMate framework (see Figure 4.1).

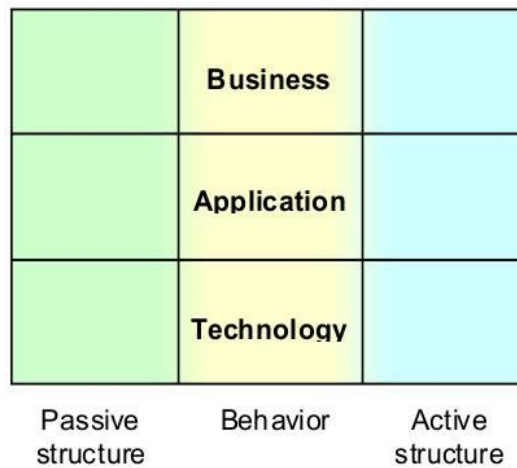


Figure 4.1. ArchiMate Framework adapted from [48]

ArchiMate modeling language consists of 3 main types of elements: active structure elements, behavior elements and objects (passive structured elements). Active structured elements are business actor, application concepts and devices. They are designed to show the elements, which can perform the actions (behavior). Behavior elements show the activity which could be performed within an enterprise. Objects are elements on which the behavior is performed.

ArchiMate could be also presented from layer perspective. There are three layers: business, application and technological. Business layer shows business processes, which bring products and services to external customer. Application layer provides different kind of application software and services, which support the business process from business layer. Technological layer mainly provides the structure of hardware of the system, which supports upper layer. However it could also have some software representation, if it supports application and business layers. Each layer of the ArchiMate model consists of different elements, which describe the behavior of this layer [48].

4.3 Illustrated example

For contribution of the proposed method CoCoME is taken under consideration. We assume that before starting with proposed algorithm (see Chapter 5) of risk assessment, the ArchiMate model of the whole enterprise architecture is made and it covers all IS and business assets of an enterprise. To constrict the scope for the method implementation, we will take ArchiMate Server Room example based on CoCoME for further contribution. General ArchiMate model of Server room example base on CoCoME is presented on Figure 4.2.

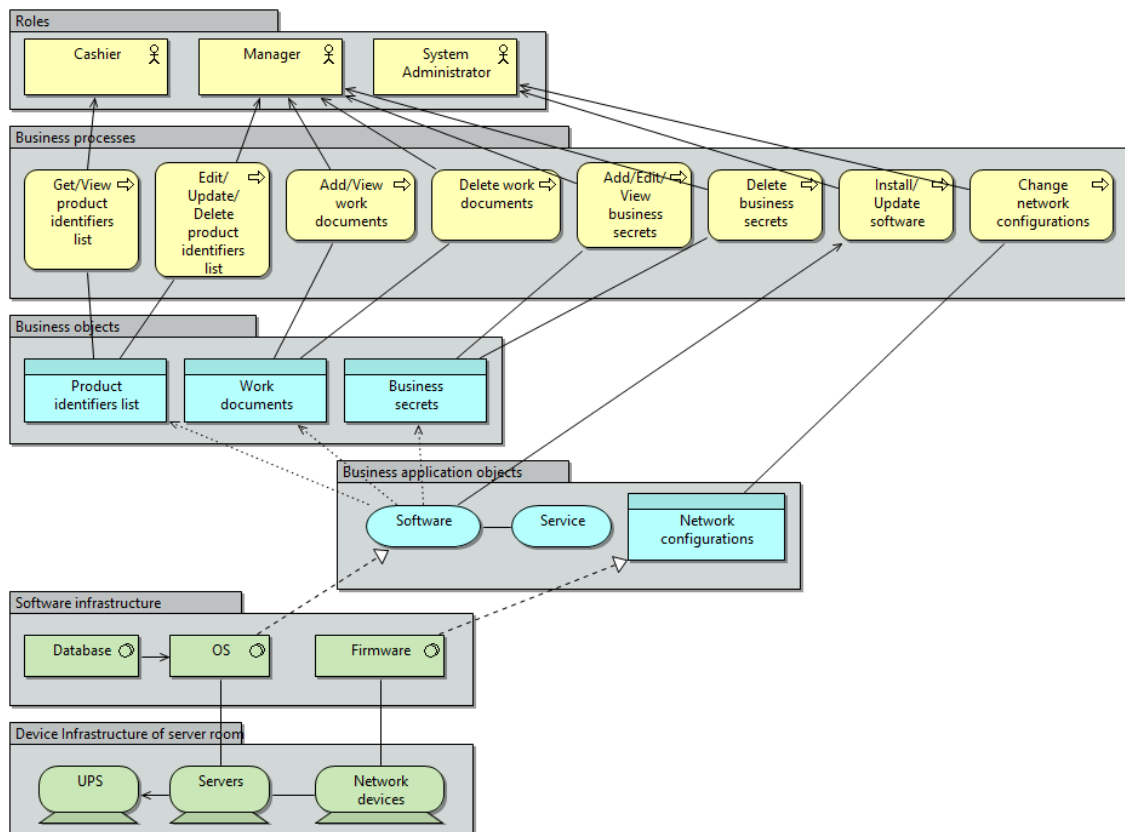


Figure 4.2. EA model of Server Room example built with ArchiMate

Architecture model of server room presented on Figure 4.2 covers important hardware, which is used for maintenance of the business flow. Since this hardware is situated in the server room, the name of example is Server Room example. The hardware is supported by software, which is presented in the *Software lane*. Both *Device infrastructure of server room* and *Software infrastructure* lanes defines Technology layer of the enterprise. Only 3 people from current enterprise can operate with presented devices and they are presented in the *Roles* lane. *System administrator* can control all processes like *Install/Update software* and *Change network configurations*. *Manager* has more rights to operate with the system (*Edit/Delete/Update PIL*, *Add/View/Delete work documents*, *Add/Edit/View/Delete business secrets*). *Cashier* has only special privileges (*Get/View PIL*) that he could not use them to misuse the system. In addition to *Roles* lane Business layer has *Business processes* lane, which presents the actions, which could be done on *Business objects* by *Roles*. *Business objects* and *Business application objects* are defined on Application layer of an EA model. This layer is a link between Technology and Business layer and defines business assets, which should be protected.

4.4 Mapping of ISSRM and ArchiMate

The alignment of RM and EAM concepts is made through development of integrated metamodel [18]. The metamodel is built using main terms from 3 ISSRM concepts. A concept mapping introduces a correspondence between at least one concepts of each of the source models. A relation between two concepts could be presented in different ways, such as a generalization, a composition, an aggregation, an association, a classification. Equivalent concepts are integrated through an alignment rule (merge, mapping, abstraction). Different connection rules (generalization, aggregation, composition, association, classification) help to integrate related rules. Once the concepts are mapped, the rules (how) to integrate the concepts within the integrated metamodel are defined [18]. To present the current alignment it is necessary to use Motivation extension of ArchiMate modeling language, as there are no elements among standard ones that can present risk-related and risk treatment-related concept of ISSRM.

The mapping from Grandry *et al.* (2013) based on the same example, which was shown in Figure 4.2 (see Figure 4.3). The model is built around *Product Identifiers List (PIL)* business asset and risk that could occur during performing operations with PIL. The main security objective of *PIL* is *Integrity of it and operation related to it*. It is presented in *Driver* element in Figure 4.3. That's why the risk for PIL business asset is *Change data in PIL*, which is presented through *Assessment* element. An impact that negates PIL's integrity is *Wrong query to PIL* and it is also defined through *Assessment*. According to chain of impacts *Wrong query to PIL* leads to *Wrong calculations for the system* and *Loss of customer loyalty*. *Software*, as business asset that is connected to PIL, has vulnerabilities that make risk occurrence possible. These vulnerabilities could be defined as *Misusage of authority to get access* and *Session duplication allowance*. The threat (*Identity theft*) and defined vulnerabilities lead the risk event (*Entrance of malicious query*). All elements from risk-related concept of ISSRM (apart security criterion) are presented through *Assessment* element.

Risk treatment-related concept of ISSRM is presented though 2 additional elements: Requirement, which shows Security Requirements and Goal, which shows Risk Treatment. To mitigate defined risk, *Enable event monitoring mechanism* and *Session duplication disallowance* are presented as risk treatment. Risk requirements are defined through *Implementation of event monitoring mechanism* and *Disallow session duplication*. They are connected to the controls, which are presented in new separate lanes: *Security processes* and *Security business objects*. *Security business objects* lane provides model with *Security objects* element, where all new elements required for risk treatment are defined. *Security processes* lane consists of processes, which help to operate with *Security objects*.

4.5 Summary

In this chapter different approaches for EA Management were presented. We select ArchiMate modeling language for further contribution as it provides structured information visualization. ArchiMate three-layered separation is not as complex as it is in Zachman framework, which makes it easier to build. ArchiMate modeling language is the most suitable for this research as it has alignment with previously chosen ISSRM domain model. This alignment is useful for the further contribution. The usage of ArchiMate was presented on the illustrated example. Moreover the application of Grandry *et al.* (2013) approach is also presented in this chapter.

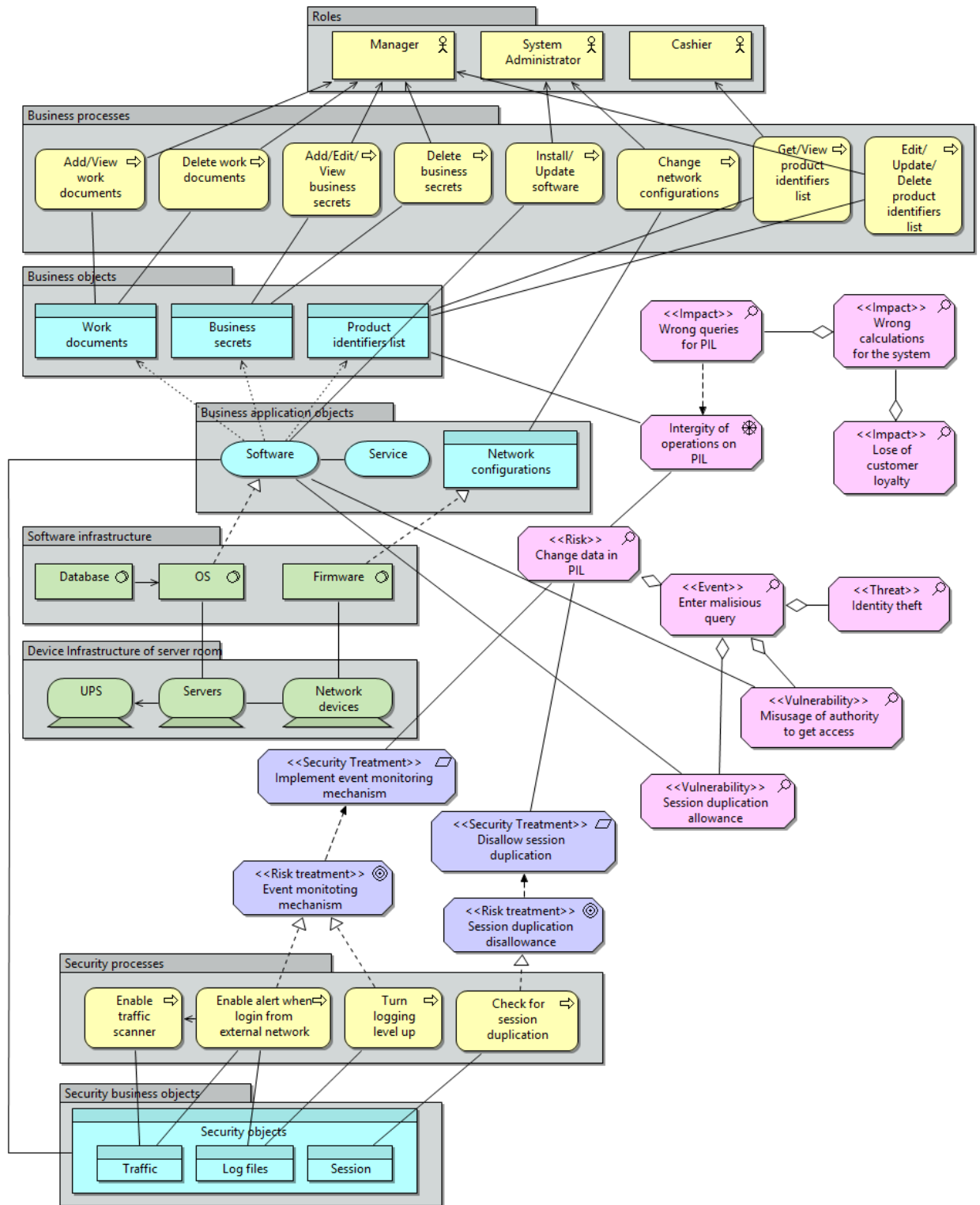


Figure 4.3. EA model of Server Room example based on Grandry *et al.* (2013) approach

5 Alignment of Enterprise Architecture and Mal-activities

5.1 Method overview

To maintain the security level of the enterprise, its architecture must contain controls, which mitigate the risk occurrence and negate vulnerabilities. It is difficult to implement *all* possible countermeasures in a scope of one enterprise, as it will be costly. That's why it is necessary to identify assets, the violation of which brings the greatest loss, or which are the most valuable for the enterprise. The step-by-step algorithm, how to make risk assessment, is presented in Figure 5.1.

Through development of the EA model it is visible which assets has company already obtained and how are they connected between each other. ArchiMate is a chosen modeling language to build the required model. Moreover to the fact that it shows assets hierarchy, the processes behind these assets are also presented in this model. Roles and actors define who operates the system and which particular assets are under whose control.

After EA model is finished it is necessary to identify the assets that must be protected. If there are no security controls implemented, all assets are needed to be taken under consideration one by one. As soon as vulnerable asset is identified, the risks that are related to this asset should be analyzed during the next step. This step could be done though drawing mal-activities diagrams, which will show how the asset could be attacked. Implementation of countermeasures also is shown in mal-activities diagrams.

Next important step is returning from implementation of countermeasures of particular asset to building them in the overall EA. To see the influence of such additions, it is useful to add already created ArchiMate model with discovered controls. They could enhance the EA model through adding new assets. Since EA model is changing after each time of method implementation, the risk analysis process should be redone considering all additions and changes.

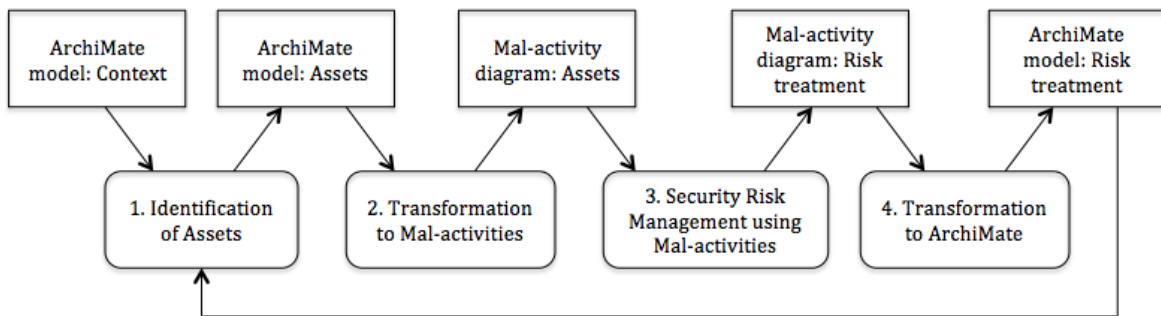


Figure 5.1. Method algorithm diagram

For method implementation we assume that analyst have whole EA model presented with ArchiMate. Apart alignments that will be define later the main rules in sequential order are presented here:

1. Separate from the general ArchiMate model only those elements, which are connected to the chosen possibly vulnerable business asset and create new model. Moreover not only direct connections must be taken into account, but also connections, which occur through layers or other elements. This step is described more detailed in Chapter 5.2.2.

2. Transform model from step 1 into ISSRM asset-related activity diagram:
 - a. Names of the roles go to names of swimlanes;
 - b. Names of IS assets go to names of swimlanes. Only directly connected IS assets are taken into account on the first loop. If it is necessary to show the additional connections, they could be transferred into this diagram as soon as they become needed.
 - c. Business processes are used in the swimlanes to show the workflow and express the connection between the swimlanes. The business process could appear only in that swimlane, the name of which is role's name to which this business process is connected in ArchiMate model.
 - d. The processes for swimlane with the IS assets name could be not found in the ArchiMate model. They should be added at the discretion of the person who is applying this algorithm.
3. Create of ISSRM risk-related MAD through adding activity diagram:
 - a. Add "Attacker" swimlane and attack methods swimlanes;
 - b. Specify the actions under "Attacker" swimlane, that attacker could make in order to violate the system;
 - c. Define the influence of attackers actions on the actions defined in other swimlanes.
4. Create ISSRM risk-treatment MAD through adding risk-related diagram:
 - a. Add "Security module" swimlane;
 - b. Specify the actions that should be done in order to mitigate the possibility of violation into the system;
 - c. Define the influence of actions specified under "Security module" on the actions defined in other swimlanes.
5. Transfer risk-treatment MAD to ArchiMate model:
 - a. The information from "Security module" swimlane from MAD should be analyzed in order to separate the new IS assets from new business assets;
 - b. If there are new IS assets, they should be put into Technical layer lane. New IS asset should be transferred into Technical layer of ArchiMate model. The connection between the new IS asset and business assets (new or existing) should be defined;
 - c. One more lane of elements should be implemented on the business layer – "Security processes". This lane will contain all processes determined in the "Security module" of risk-treatment MAD.
 - d. From each of these processes define the business object that they use and add these business objects into "Security business objects" lane of ArchiMate model from step 2.
 - e. Make suitable connections between elements.
 - f. The connection between process and person (people), who will perform this operation, should be specified.

5.2 Identification of Assets to Protect

5.2.1 ArchiMate Alignment to ISSRM: Asset Model

In terms of ISSRM asset-related concept, ArchiMate model specifies the IS and business assets. Although it contains information, which is required for building the asset-related MAD, there is no security criterion mentioned.

ArchiMate model proposes many elements to describe the EA. However the names of the elements differ from the ISSRM terms. To make risk assessment it is needed to map ArchiMate and ISSRM terminology. The mapping was done through analysis of the elements descriptions of both concepts. The alignment of the elements, which were used in our example, is presented in Table 5.1. The validity of alignment is supported by example description based on Server Room example. *Business process as element of ArchiMate could be mapped to business asset from ISSRM DM*, since Get/View PIL is an element, which defines to operations with *PIL* and describes the process essential to the business. *Application service and Data object elements from Application layer of ArchiMate could be also mapped to Business asset element of ISSRM DM*. The examples are: Software is service that shows automated behavior and describes the processes essential to the business; *PIL* is a passive element, which describes the information essential to the business and is suitable for automated process. *Device and System software elements from ArchiMate Technology layer could be presented through IS asset of ISSRM DM*. The validity is shown on examples: Server is a hardware resource, which stores or deploys for execution *PIL*, word documents, and business secrets in order to support business assets, which are defined in business process lane; OS is a software environment for deployment of *PIL*, word documents, business secrets in order to support business assets, which are defined in business process lane. Unfortunately *there are no elements of ArchiMate modeling language that could be aligned to Security criterion element of ISSRM DM*.

The number of listed elements is enough to build simple EA model as it is done in the Figure 4.2. If it is needed to add the mapping for more elements, the analysis should be done in the same way. The approximate mapping is presented in the Table 5.2. The element alignment could vary depending on the particular example.

Table 5.2. ArchiMate and ISSRM asset-related concept: general alignment

ISSRM	ArchiMate	
IS asset	Node, Device, Network, Communication path, Infrastructure interface, System software, Infrastructure service, Artifact	<i>Technology layer</i>
	Application component, Application collaboration	<i>Application layer</i>
Business asset	Application interface, Application service, Data object	<i>Application layer</i>
	Business collaboration, Business interface, Business function, Business interaction, Business event, Business service, Business object, Meaning, Value, Product, Contract	<i>Business layer</i>

Technical layer of ArchiMate gives an information about IS assets, which are used by the system to maintain the work process and support business assets. Although this level could be defined only over hardware devices, it is also possible to add the separate block with software. This mainly could be done for more clear separation between application and technical layers. Specification of the business assets is made on the application layer. Business processes show the actions, which could be done with assets. Business processes also help to make the connection between the business assets and roles (people who operate with these assets). Roles are not aligned to any of ISSRM terms.

5.2.2 Asset Identification Example

Application of previously proposed alignment to the Figure 4.2 gives us the Table 5.3. Implementation of proposed rules presents *Device* elements (Servers, Network devices and UPS) as IS asset. According to the alignment and Server Room example IS assets also

could be taken from System software (Database, OS, Firmware). Business assets could be presented through Application service, Data object or Business process elements. From Server Room example it is visible that *Application services* are Software and Services. Network configuration, PIL, Work documents and Business secrets are presented through *Data object* elements. *Business process* elements are used to define Get/View product identifiers list, Edit/Update/Delete product identifiers list, Add/View work documents *etc.*. Although each layer of the EA model could have vulnerabilities, for further contribution we will take the business layer. Since business assets from this layer are based on the assets from bottom layers, all improvements made for this layer have influence on the related elements through whole EA model.

Table 5.1. ArchiMate and ISSRM asset-related concept alignment

ISSRM definition	ISSRM DM	ArchiMate elements		ArchiMate definition
An element which describes the information, processes, capabilities and skills essential to the business and its core mission.	Business asset	Business layer	Business process	A behavior element that groups behavior based on an ordering of activities. It is intended to produce a defined set of products or business services.
		Application layer	Application service	A service that exposes automated behavior.
			Data object	A passive element suitable for automated processing.
The IS component, valuable to the organization since it supports business assets.	IS asset	Technological layer	Device	A hardware resource upon which artifacts may be stored or deployed for execution.
			System software	A software environment for specific types of components and objects that are deployed on it in the form of artifacts.
The property or constraint on business assets describing their security needs, which are, typically, expressed through <i>confidentiality</i> , <i>integrity</i> and <i>availability</i> [38].	Security criterion	-		

In case of Server Room example, which EA model is presented on Figure 4.2, business assets are defined in the *Business processes* lane: Edit/Update/Delete PIL, Get/View PIL, Install/Update software and Change network configurations *etc.*. Each of these elements from that lane should be taken under consideration, what makes the application of proposed method gradual. In other words, the method should be applied to each of the elements which present business asset on the business layer on the ArchiMate model.

The algorithm of risk assessment will be done based on the *PIL* as a business asset. Firstly, it is necessary to distinguish the elements, which are related to the chosen business asset. The relation could be identified not only through direct link, but also through layers. After the required elements and relations were identified, they should be separated in new EA model. This model is presented in the Figure 5.2.

Table 5.3. ArchiMate and ISSRM alignment based on Server Room example

ISSRM	ArchiMate Element name	Server Room ArchiMate model	
IS asset	Device	Servers	<i>Technology layer</i>
		Network devices	
		UPS	
	System software	Database	
		OS	
		Firmware	
Business asset	Application service	Software	<i>Application layer</i>
		Services	
	Data object	Network configurations	
		PIL	
		Work documents	
	Business process	Business secrets	
		Get/View product identifiers list	<i>Business layer</i>
		Edit/Update/Delete product identifiers list	
		Add/View work documents	
		Delete work documents	
		Add/Edit/View business secrets	
		Delete business secrets	
		Install/Update software	
Change network configurations			

The direct connections between *PIL* and its business processes (*Edit/Update/Delete PIL* and *Get/View PIL*) determine the connections to the *Cashier* and *Manager* in the Roles lane. Defined business asset has a relation link to *Software*. This connection gives the opportunity to follow the link to the IS assets which support *PIL*. Since there is a link between *Software* and *Install/Update Software*, the *System Administrator* role should be also transferred to the new ArchiMate model. All connections from *System Administrator* role should be also presented in the new ArchiMate model, even if they do not have direct connection to the chosen business asset.

The transformations, which should be done in order to continue with risk assessment through proposed algorithm, could be summarized in following rules:

1. Identify from Application layer the business asset, on which all further analysis will be based;
2. Define the elements, which are related to the chosen business asset. Elements relation could be defined through layers;
3. Transfer chosen business asset and elements, which are related to it into separate EA ArchiMate model.

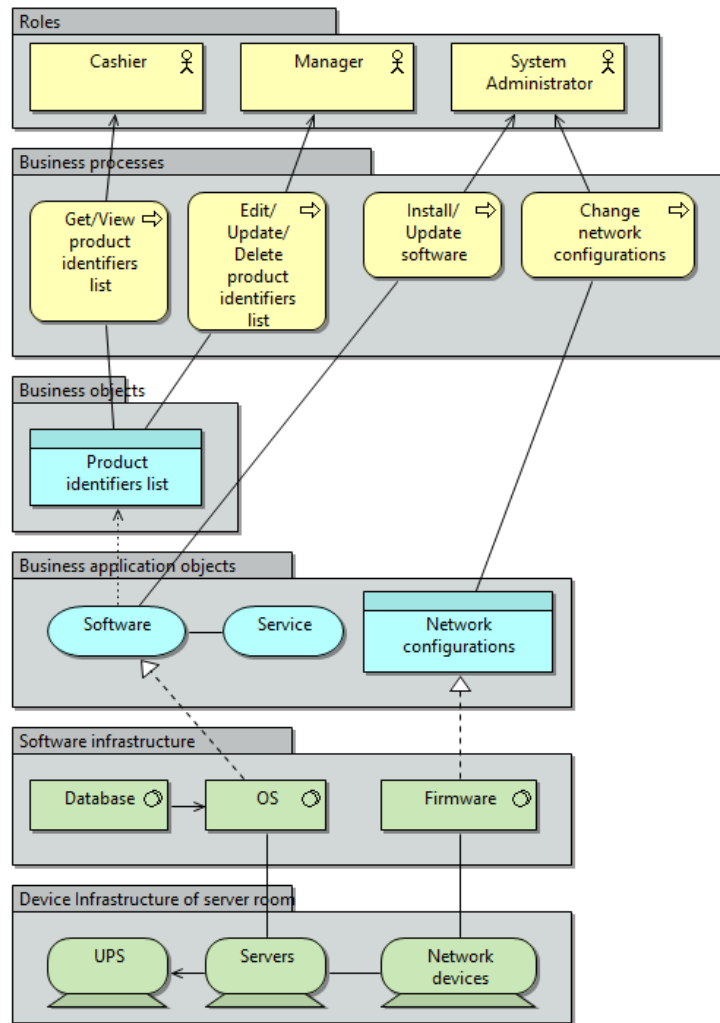


Figure 5.2. EA model of Server room example defined for chosen business asset

5.3 Transformation to Mal activities

5.3.1 Transformation rules

For further steps in risk assessment we need to make risk analysis. It could be done with the help of modeling language, which, in our case, is MAD. Unfortunately, the elements from ArchiMate modeling language and MAD are not the same, so it is necessary to make a mapping between them, that we can use the knowledge from the EA in the risk analysis.

The link between these two languages is ISSRM concepts. The mapping of ArchiMate elements to the ISSRM terms partially was made in Chapter 5.2.1. The alignment between MAD and ISSRM is shown in the paper presented by Chowdhury *et.al.* [8]. The combination of these two alignments gives the general set of mapping rules between MAD and ArchiMate. The summarized alignment is presented in the Table 5.4. This mapping is done only for elements, which are used in Server Room example. If there were more elements in use in ArchiMate EA model, the alignment should be done in the same way. The mapping for ISSRM asset-related concept is presented in Table 5.4. Device and System software, as elements of ArchiMate, could be presented though swimlane in MAD. MAD elements such as Activity and Decisions, which are connected using ControlFlow

constructs, could present Application service, Data object and Business process from ArchiMate elements.

Table 5.4. Alignment between ArchiMate and MAD: asset-related concept

ISSRM	ArchiMate Element name		MAD
IS asset	Device	<i>Technology layer</i>	- Swimlane
	System software		
Business asset	Application service	<i>Application layer</i>	- Activity, Decision (connected using ControlFlow constructs)
	Data object		
	Business process	<i>Business layer</i>	

5.3.2 Transformation example

The outcome of the first step in proposed Server room example is shown in the Figure 5.2. The application of the proposed transformation algorithm should be applied only to the elements, which are directly connected to a chosen business asset. As it was mentioned before, the *PIL* is the business asset on which the whole example analysis is based on. The alignment in combination with proposed rules for chosen example is presented in the Table 5.5. Server is presented through ArchiMate Device element, which after transformation into MAD becomes a Swimlane with name Server. Request to View/Get *PIL* and Request to Edit/Update/Delete *PIL* are shown as Activity elements of MAD and present Business process ArchiMate element. In Figure 5.3. business process, which are used in this example, are defined as View/Get *PIL* and Edit/Update/Delete *PIL*. To make process more precise before this operation will be completed, the request for this action should be done.

Table 5.5. Alignment between ArchiMate and MAD based on Server room example

ISSRM	ArchiMate element name		MAD element name	Example
IS asset	Device	<i>Technology layer</i>	- Swimlane	Server
Business asset	Application service	<i>Application layer</i>	- Activity, Decision (connected using ControlFlow constructs)	-Request to View/Get <i>PIL</i> ; -Request to Edit/Update/Delete <i>PIL</i> ; - View/Get <i>PIL</i> ; - Edit/Update/Delete <i>PIL</i> .
	Data object			
	Business process	<i>Business layer</i>		

The roles, which have direct connections (even through layers), are transferred into the name of the swimlanes: *Cashier*, *Manager* and *System administrator*. *Server* is one more swimlane, which is added, in asset-related diagram, since it also has direct through-layered connection with the *PIL*. There are many different actions behind every business process. That's why to get the required business process done, it is necessary to start with request for doing this process. It is visible from Figure 5.3. There are activities *Request to View/Get PIL* and *Request to Edit/Update/Delete PIL*. After the sequence of processes from server part is completed, the roles (*Cashier* and *Manager*) will get the process *View/Get PIL* and *Edit/Update/Delete PIL* respectively. The person who is doing the risk analysis should think through the set of actions, which are done from the server side, in order to serve the requested process. This information is not presented in the EA model.

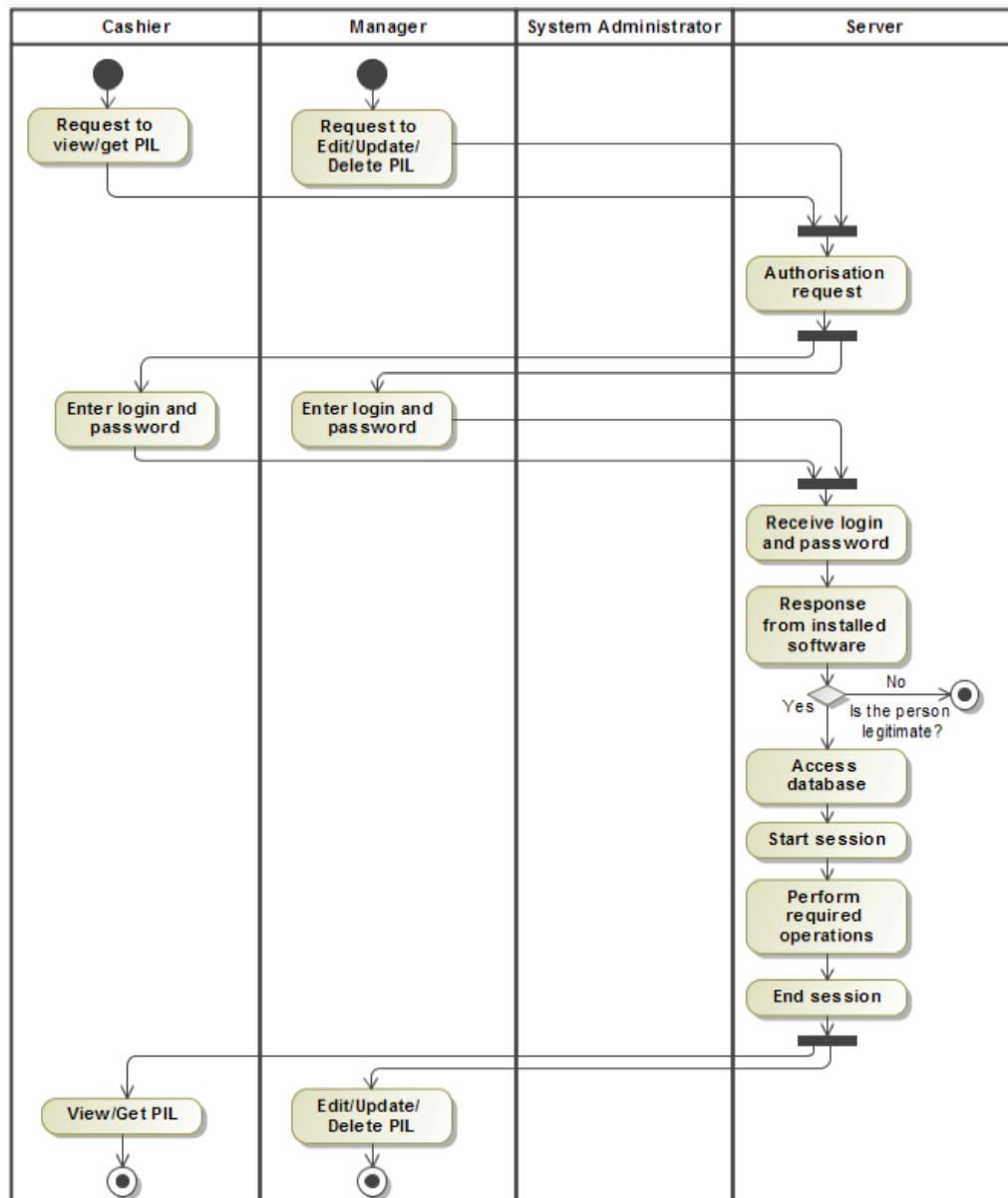


Figure 5.3. MAD asset-related model for Server room example

5.4 Security Risk Management using Mal-activities

The further analysis of the asset-related concept should be done through basic MAD algorithm. The next diagram that should be made is risk-related MAD. With the help of this diagram the attack algorithm is presented. After vulnerable points, where the risk can occur, are defined, risk-treatment diagrams should be built. It contains “Security module” that defines security control and shows the ways, how the risk could be mitigated.

The asset-related MAD diagram is transferred to the risk-related diagram without any changes. During diagram building, analyst should think through on which level the risk could occur and which process could influence on its occurrence. The attack could be planned beforehand, how it is shown in the Figure 5.4.a. and 5.4.b. *Attacker* sends *email* to different people from the system (*Cashier* and *Manager*). *Attacker* also could send an email to *System administrator*, but it will not have any influence on PIL, as *System administrator* does not operate with PIL. The system is built in a way that *Cashier* does not have Internet access on his/her work computer. That’s why if *Attacker* applies social

engineering and sends email to *Cashier*, *Cashier* could open it only on his/her own private computer and it will not influence on the system. However *Manager* has his/her email account on the work computer, which makes the attack method through *sending email with malicious attachment* available. If the attack was successful and *Manager* opened email on his/her work computer, the *installation of the keylogger* starts. We assume that there are no countermeasures preinstalled which can prevent installation. After installation is finished, keylogger starts to monitor all the input of the *Manager*. Eventually the *Manager* will try to access the database with PIL through special *software*, which requires authentication. As soon as he/she will enter his/her credentials, the keylogger will get it. The *Attacker* just needs to analyze the data and as soon as he/she will see the required for him/her information, send it to him/her. After the *Attacker* obtains the credential for access the database, he/she can *duplicate the session*, enter the obtained credentials and get access to the confidential documents (PIL). We assume that the goal of the attacker is to mess up the PIL. That's why as soon as he/she gets access to the database, he/she *create a request to insert malicious data into a database*. It becomes possible through SQL injection.

Secondly, after risk-related diagram is built, analyst should think about countermeasures that will help to prevent the risk occurrence. Risk treatment-related concept is presented on Figure 5.5.a and 5.5.b "Security module" is implemented for these purposes. Different countermeasure actions are presented in "Security module" lane. The first implemented control is *Enabled email filter and antivirus*. It influences on email delivery. If the control is implemented correctly, it should monitor the *malicious attachments in emails*. That's why the *Manager* should not *Receive the email with malicious attachment* at all. However if it happened, there are more controls to detect the malicious activity. If *Manager* still received the *email with malicious attachment* and opened it, *The silent installation mode of keylogger* should not be allowed. In other words, only *System Administrator* must have right to *install the software to employee's computers*. Even if the keylogger was installed, it should not be able to send the obtained information. This could be mitigated through *traffic scanner*. If the credential were obtained through another place, the misuser should not be able to *duplicate the session*. Usually attacker is working from the external network, so there should be *alert turned on in order to detect the violation from the external network*. All changes and manipulations with the database should be *logged in the log files*.

5.5 Transformation to ArchiMate

5.5.1 ArchiMate Alignment to ISSRM: Risk treatment-related concept

The risk-related concept is made on the basis of ISSRM asset-related concept. Since there is no data, which could be transferred from ArchiMate to risk-related concept, there is no need for alignment between this concept and ArchiMate. The next mapping that is required for further analysis is alignment between ArchiMate and ISSRM risk treatment-related concept (see Table 5.6). This mapping will be used on the last stage of the algorithm.

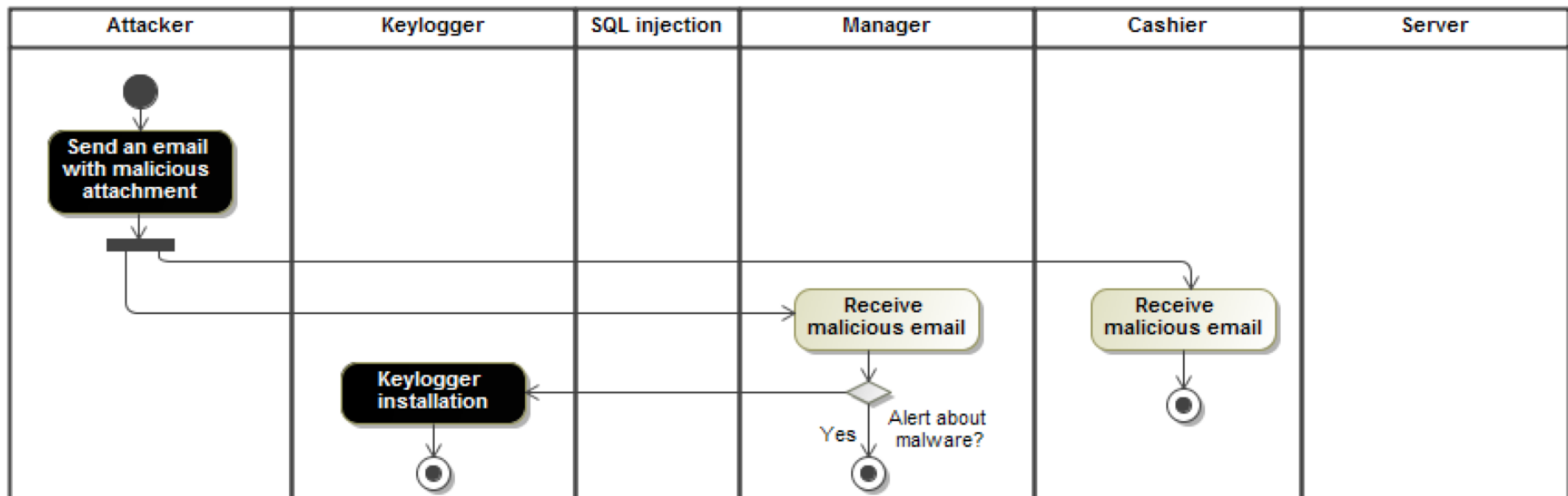


Figure 5.4.a. MAD risk-related model for Server Room example

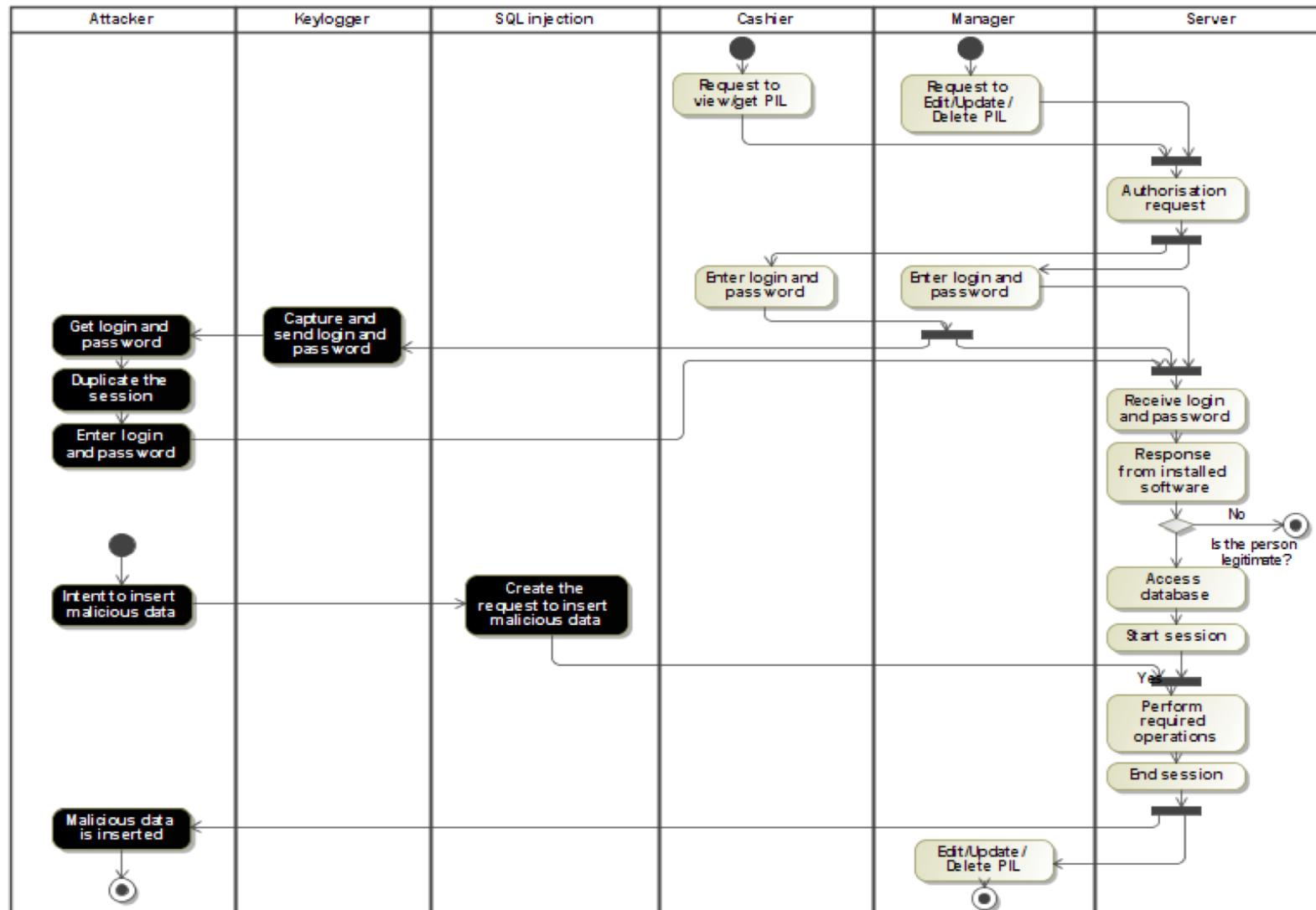


Figure 5.4.b. MAD risk-related model for Server Room example

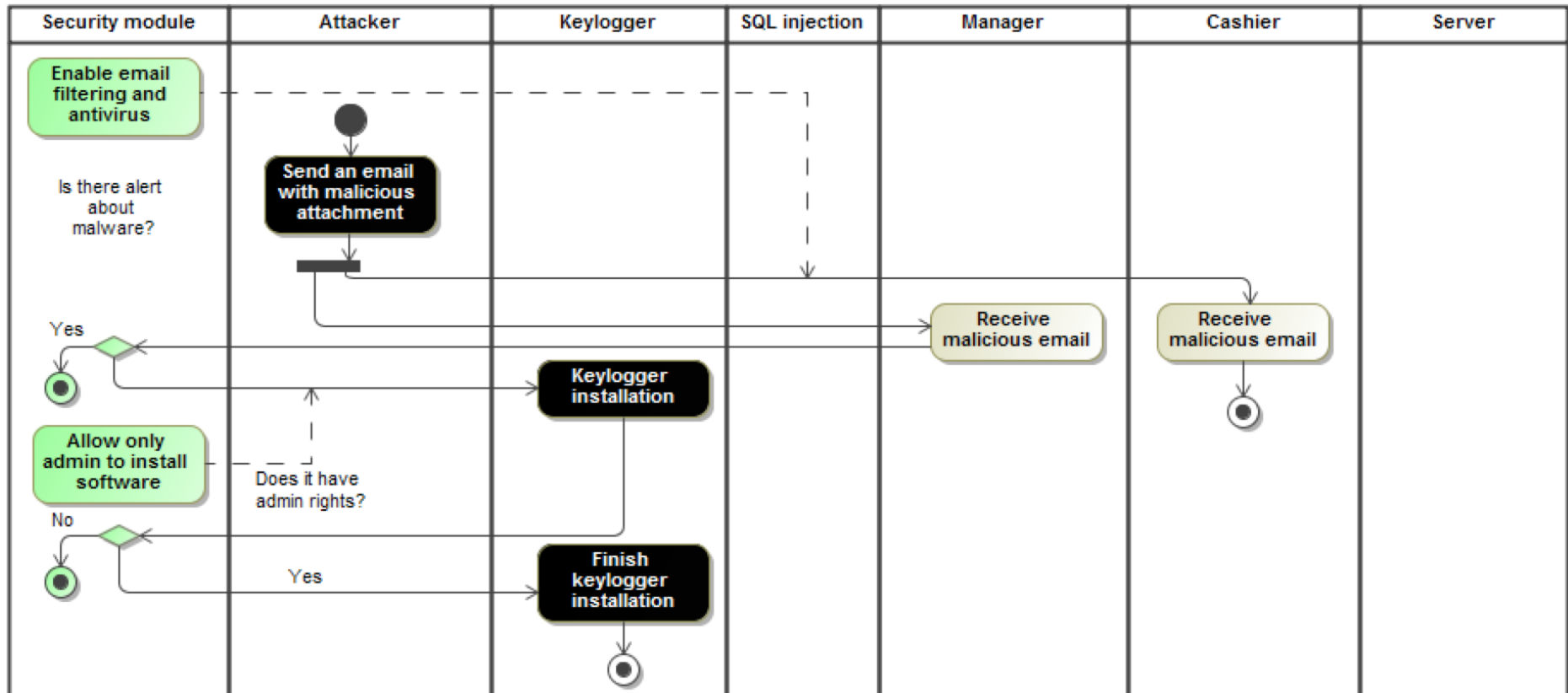


Figure 5.5.a. MAD risk treatment-related model for Server Room example

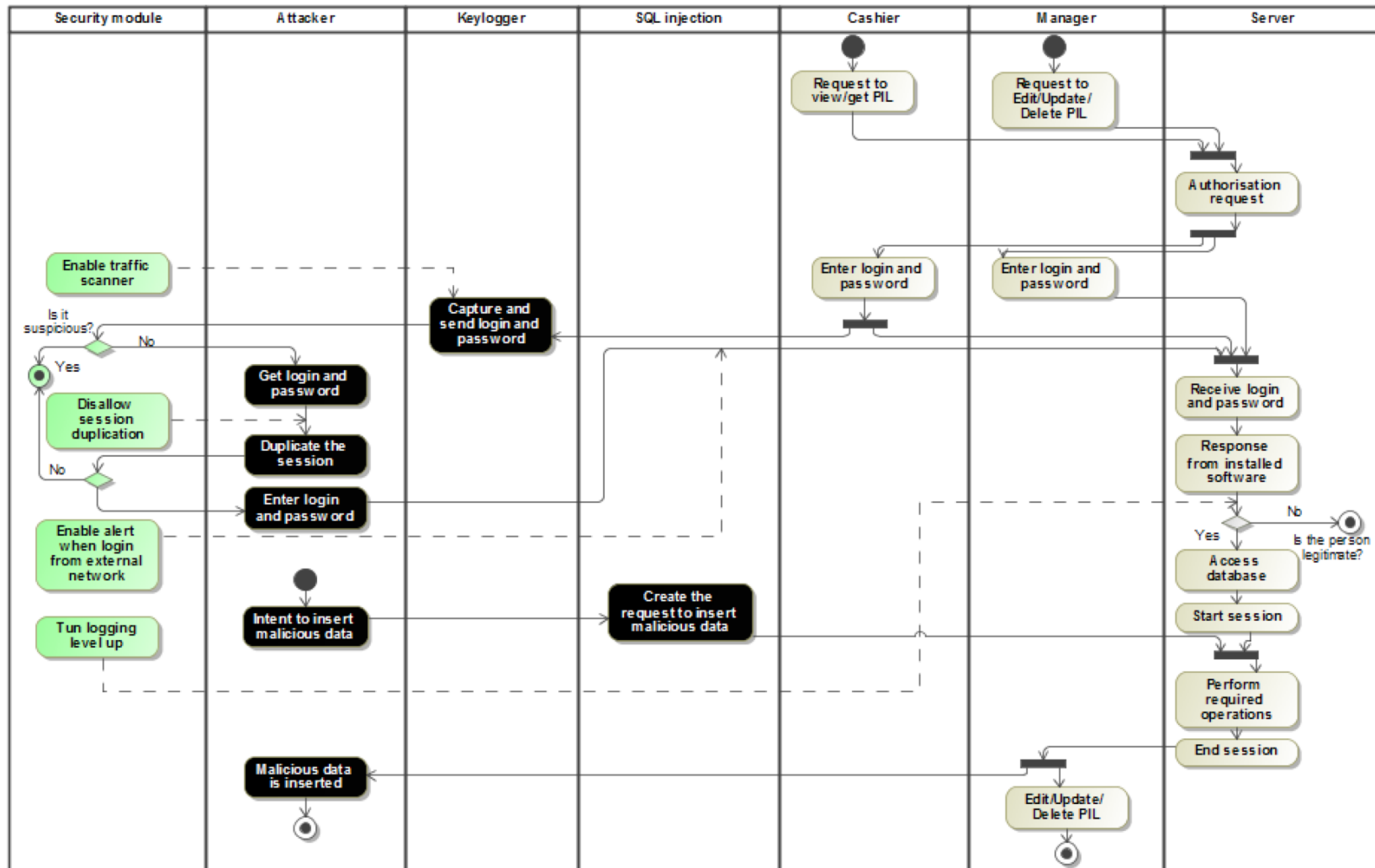


Figure 5.5.b. MAD risk treatment-related model for Server Room example

Security requirement as element of ISSRM DM is aligned to the Business process ArchiMate element. Proving the right of mapping could be done through definition of elements from Server Room example. *Check session duplication* is an element that shows the behavior based on ordering of activities (operating with business assets) and provides the condition that session should be checked on its duplication in order to mitigate risks. Control element of ISSRM DM could be presented through 2 ArchiMate elements. One is Data object from Application layer and second is System software from Technology layer. The examples are as following: *Session* is a passive element suitable for automated process and it is designed to improve security specified by a security requirement (the session should be checked on its duplication); *Traffic scanner* is software for specific types of objects (traffic) and it is design to improve security in a way of monitoring the traffic. Since it is difficult to show risk treatment decision in visual concept, there is no construct in ArchiMate determining risk treatment element from ISSRM DM.

Table 5.6. ArchiMate and ISSRM risk treatment-related concept general alignment

<u>ISSRM definition</u>	<u>ISSRM DM elements</u>	<u>ArchiMate elements</u>		<u>ArchiMate definition</u>
A condition over the phenomenon of the environment that we wish to make true by installing the IS [12].	Security requirement	Business layer	Business process	A behavior element that groups behavior based on an ordering of activities. It is intended to produce a defined set of products or business services.
A designed means to improve security, specified by a security requirement and implemented to comply it [12].	Control	Application layer	Data object	A passive element suitable for automated processing.
		Technological layer	System software	A software environment for specific types of components and objects that are deployed on it in the form of artifacts.
A decision of how to treat the identified risk [12].	Risk treatment			

5.5.2 Transformation rules

The last step of the algorithm requires transformation back from MAD to ArchiMate model. On this step MAD presents ISSRM risk treatment-related concept. Alignment for this concept, based on the chosen example, is presented in Table 5.7. Decision and Mitigation activity combined using control flow as MAD elements could show Business process, Data object and System software elements of ArchiMate. Unfortunately there are no constructions in ArchiMate to present Swimlane, which shows Control element of ISSRM DM.

Table 5.7. Alignment between ArchiMate and MAD for risk treatment-related concept

ISSRM	MAD	ArchiMate Element name	
		Security requirements	Decision and Mitigation activity combined using control flow
Data object	<i>Application layer</i>		
System software	<i>Technological layer</i>		
Control	Swimlane	-	
Risk treatment		-	

5.5.3 Risk treatment in ArchiMate

After risk assessment was made, it is necessary to make the reverse transformation from MAD diagram, which contains controls and countermeasures, to ArchiMate to see, on which layer the controls will occur. The implementation of the countermeasures in the EA shows on which assets will these controls influence and the connections between already existed and new assets. The alignment for application of this transformation is presented in Table 5.6. The implementation of these mappings on Server Room example is defined in Table 5.8.

Table 5.8. Alignment between MAD and ArchiMate based on Server Room example

ISSRM	MAD	ArchiMate Element name		Example
Security requirements	Decision and Mitigation activity combined using control flow	Business process	<i>Business layer</i>	Send/Receive email
				Enable email filtering
				Allow only to install software
				Enable traffic scanner
				Check for session duplication
				Enable alert when login from external network
				Turn logging level up
		Data object	<i>Application layer</i>	Email
				Permissions
				Session
System software	<i>Technology layer</i>	Traffic		
		Log files		
Control	Swimlane	-		
Risk treatment		-		

According to the Server Room example the mapping could be implemented like this: Business processes like Send/Receive email, Enable email filtering, Allow only to install software *etc.* are presented as Decision and Mitigation activity elements of MAD combined using control flow. Same MAD elements also could define Data objects elements like Email, Permissions, Session *etc.* and System software element (Traffic scanner).

Transferring the alignment, presented in Table 5.8, into the actual model is not so easy, as analyst should think carefully, where the element would appear. One more important point is that Table 5.8 does not cover the connections between the elements. This point of EA model developing is up to analyst understanding.

EA model of Server Room example after implementation of the method is presented in Figure 5.6. The main element, which was added to this model, is “Security processes” lane, which is a buffer between the business processes and business objects. Now before getting the *PIL* through *Get/View PIL* or *Edit/Update/Delete PIL*, the sequence of security process will take place. Besides “*Security processes for operations with PIL*” there are two more security processes which appear because of possible malicious action from the attacker. They present general controls, which help to keep system secure: *Enable email filtering* and *Allow only admin to install software*. Since there was no business process related with emailing process, *Send/Receive email* process was added in “*Business processes*”. “*Business objects*” lane is added with elements, which were defined through. Each security process from “Security processes” lane operates with business object. Analyst needs to think through which assets it is, and if there is none already presented in the EA model, add it in the “Business objects” lane. From Figure 5.6 it is visible that these elements are: *Email, Permissions, Session, Traffic, Log files*. *Traffic scanner* was added to “Software infrastructure” lane as it is IS asset, which supports business asset (*Traffic*).

5.6 Summary

This algorithm gives the help hints in the whole risk assessment process. However it could be modified for the particular problem of the IS of an enterprise.

The presented steps should be applied to each of the business assets. According to the algorithm the number of business assets will grow with each loop of analysis. This will happen, since after reverse transformation from MAD to ArchiMate countermeasures from MAD will be presented as business assets in ArthiMate. Eventually all business assets will be covered, so this will be a sign that full analysis was made.

The fact that analysis was made once does not give any proof that the system could not be violated. The countermeasures should be revised periodically. The implementations of new assets or extension of the business should be immediately added to the general EA model. The analysis should be repeated, if there was a breach in the security system of an enterprise.

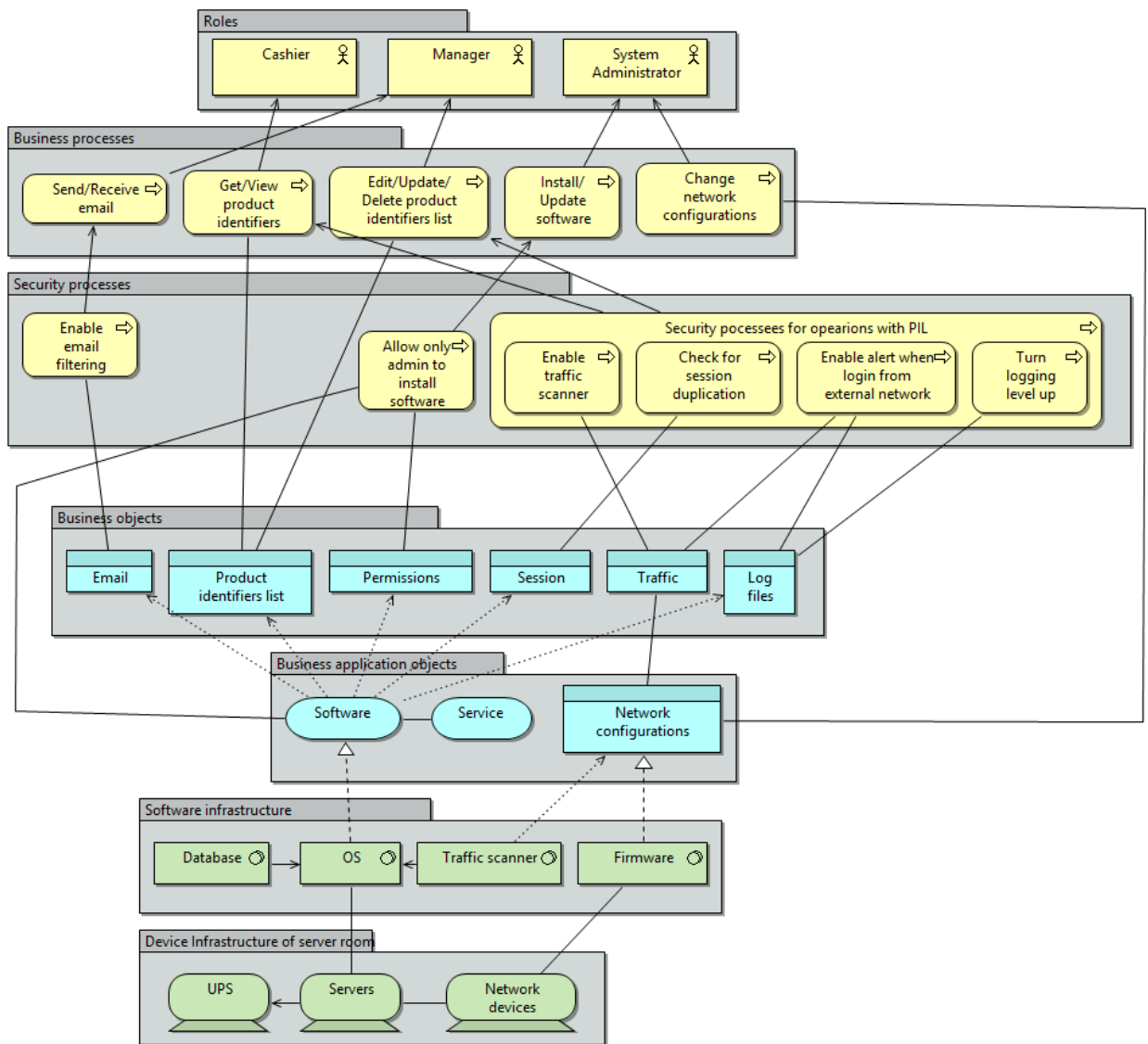


Figure 5.6. EA model of Server Room example after method implementation

6 Validation

In this chapter we validate the different concepts of alignment between EA and SRM. The original concept is presented by Grandry *et al.* (2013) and proposes the alignment through mapping ArchiMate EA model and ISSRM DM. To make the alignment full, Motivation extension of ArchiMate is required to be used. The outcome of application of this concept is presented in Figure 4.3.

The second concept, which is going to be used in validation, is concept from current work. The same as in previous concept, ISSRM DM is used as main SRM concept. ArchiMate is taken as modeling language to present EA. Unlike aforementioned approach, the method, proposed in this work, uses one more intermediate step to make the alignment more clear. For implementation of this step MAD is being used. That's why one of the outcomes of the current method is the alignment between ArchiMate and MAD. This intermediate step helps to perform risk analysis very carefully. The EA model after implementation of this method is presented in Figure 5.6.

6.1 Research question and method

The main validation question could be formulated as: *Which method of alignment between EA and SRM is more complete and precise?* To answer this question we will go through the steps presented in Figure 6.1.

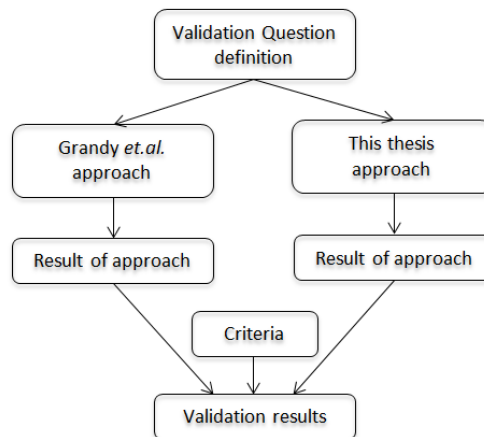


Figure 6.1. Validation steps

Approach presented by Grandry *et al.* (2013) is using ArchiMate as a modeling language to design EA model. For making risk analysis based on developed model, usage of standard ArchiMate modules is not enough. For this purpose Motivation extension of ArchiMate is being used. It helps to present elements of risk-related and risk treatment-related concepts of ISSRM. However controls and countermeasures are shown through standard modules.

Although method presented in this work is also using ArchiMate for modeling EA model, it does not require any extra ArchiMate modules. It is only necessary to make one additional alignment to present risk-related concept of ISSRM. It is done through MAD, which shows steps of attacker very precise. Risk treatment-related concept of ISSRM is also presented through MAD. It helps to implement the mitigation countermeasures on each malicious step of the attacker. That's why it gives broader picture of possible

controls, which could be implemented in the system in order to mitigate risk occurrence. The transformation back from MAD to ArchiMate gives the presentation of chosen countermeasures in the whole EA.

6.2 Summary of results

The application of both aforementioned concepts provides risk analysis and proposes countermeasures to implement into system in order to mitigate the risks. The comparison of results is done based on outcome of two methods; in particular achieved EA models (see Figure 4.3 and 5.5). We assumed that attacker wants to enter malicious data into PIL. Through approach presented in this work we can see the sequence of actions that attacker should accomplish in order to make the attack successful. Moreover in Figure 5.3 it is shown how attacker's actions influence on the normal business processes flow. In case of approach propose by Grandry *et. al.* (2013) it is only visible to which asset the risk is related and which vulnerabilities are presented in the system. *Unlike the method presented in this work, Grandry et. al. (2013) approach does not show all possible ways of attack on chosen asset.*

The main difference, which is visible from the outcome figures (see Figure 4.3 and Figure 5.6), is that Figure 5.6. also presents general controls, when Figure 4.3 covers only countermeasures related to the chosen asset. In Security processes lane in Figure 5.6 there are two more additional elements that covers general security requirements. Through risk analysis, which was made for a particular asset, it turned out that there is no email filtering and there are no permission restrictions for software installation. In case of Grandry *et. al.* (2013) approach these countermeasures are not covered, as there were no related threat and vulnerability detected. *Approach presented in this work detects more threats and vulnerabilities than Grandry et. al. (2013) approach.* Grandry *et. al.* (2013) method is more general and provides controls related only to the asset, which was chosen to protect. However the concept presented in this work shows more possible ways to attack the system, that's why it brings more controls and countermeasures definition.

Talking about complexity of visual presentation of outcome, *approach defined in this work presents EA model in more structural and understandable way than Grandry et al. (2013) method.* In order to present risk-related and risk treatment-related concepts of ISSRM, method proposed by Grandry *et. al.* (2013) requires usage of additional module of ArchiMate. Implementation of elements from this module is not structured in lanes, what makes relations and general representation more complex for understanding. If Grandry *et. al.* (2013) approach is applied to all assets of the system, the EA model will just explode (it will be impossible for person to keep track of the assets, controls and relations). Although method proposed in this work enhances after each application loop, it keeps structured representation of information. The elements could occur only within the lanes, what makes the relations and connection visible and understandable.

As it was mentioned before, approach proposed in this work does not need any additional modules from ArchiMate to present elements, which are required for risk analysis. On another hand, it needs one more alignment (between MAD and ArchiMate), which is not required in case of Grandry *et. al.* (2013) approach. This alignment separates the risks, which are related to particular asset and provides the countermeasures within the scope of chosen asset. Proposed approach makes risk analysis structured and shows all possible attacks on chosen asset. After risk analysis is done, method from current work provides the description of how countermeasures should be implemented back to the original EA and shows, how they are connected to the already existed assets and operations. In approach

proposed by Grandry *et. al.* (2013) risk analysis and countermeasures implementation are done in one model, which, in case of big example, will make model not understandable. Implemented countermeasures for one asset are not visible as risk-mitigation controls, due to not structured representation. Moreover *the usage of EA model for risk analysis (in Grandry et.al. (2013) concept) does not give a possibility to look though all possible attack scenarios, when in case of approach presented in this work all malicious activities could be designed on MAD stage.*

It is difficult to mitigate all risks, which can occur within operation with one asset, but it is necessary to take into account all of them and decide which is the most harmful. Building MAD brings sequential presentation of operations from the IS asset side and how these operation are related to business processes of the system. For understanding of IS asset operations analyst should have IT background. However in case of Grandry *et. al.* (2013) approach person, who implements the concept, just need to have good understanding of business flow and EA structure. The requirement for IT knowledge brings approach presented in this work strong application limitation.

6.3 Discussion

Based on the results from previous sub-chapter it is visible that with the help of additional alignments (ArchiMate to MAD and back) the concept defined in this work shifts the complexity of risk analysis from EA modeling language to risk-oriented modeling language. All risk analysis will be done separately based on information from EA model. Despite countermeasures decisions are also done separately from EA model, they easily could be transferred back and do not have influence on the structured representation of the EA model.

Usage of MAD in the concept, which is defined in this work, as special risk-oriented modelling language, negates the limitation of leakage of expressions in order to show the risk. ArchiMate Motivation extension, which is used in approach presented by Grandry *et.al.* (2013), does not have big variety of elements for presentation of risk-related and risk treatment-related concepts of ISSRM.

6.4 Threats of validity

One of the main threats of validity for method presented in this work is size of example. Due to lack of time the approach was applied only to one business asset and already enhanced the size of the overall model. Application of the method on the very big EA model could enlarge the model to the size that it will be difficult to keep track on it. Optimization of proposed method is not presented in this work and remains for future research. There is need to ask people to apply current method accordingly to defined rules to make us understand the weak points of proposed approach. Unfortunately, it was not done yet due to lack of time.

One more limitation of this research is subjectivity. The subjectivity of analyst, who is applying proposed approach, could influence on the outcome of the method implementation. There are some points in the concept rules, which are left for analyst definition. That's why his/her understanding of the whole EA and business process in particular enterprise will influence on the outcome of risk analysis.

Last but not least threat of validity is availability of information to perform risk analysis. Approach, presented in this work, helps to align business and IT part of enterprise. If the

analysis will be done by business related person, he/she could be just lack of IT-related information which is important to fulfill the alignment of presented concept. On another hand, IT-related analyst could not take into account risks related to business process. That's why there should be good cooperation between experts from business and IT field to provide each other with required information for risk analysis.

6.5 Summary

The validation was based on the Server Room example. The outcome models of Grandry *et. al.* (2013) concept and approach defined in this work were compared. Despite of the number of the additional models that should be designed in order to make risk analysis, approach presented in this works gives more complete result. Since it separates the risk analysis from EA modeling, it considers all possible attack methods that could be used against the system. Unlike the Grandry *et.al.* (2013) method, concept, defined in this work, maintains the structure representation of EA model and keeps it understandable for analyst.

7 Conclusions and future work

In this work the method for alignment between the EA and SRM was defined. It was done through mapping of ArchiMate EA modeling language and MAD risk-oriented modeling language. The validation was based on comparison of the outcome models, archived after application of defined method and Grandry *et.al.* (2013) approach on Server Room example. The results of validation showed that despite the necessity for designing additional models for risk-analysis, the concept defined in this work presents more complete and precise outcome model.

In this chapter the limitation of this work will be discussed. The conclusions and research question answer are also defined here. Last but not least, the recommendation for future work will be presented in the end of the chapter.

7.1 Limitations

The one of the limitations of this work is subjectivity. The rules and guidelines, which are defined in approach presented in this work, are based on our understanding of EA model. The example chosen for illustration of the method also made adjustments of the rules application. Thus, it might mean that some aspect of alignment between MAD and ArchiMate could be interpreted differently. The correction in mapping also could be based on the specific example to which the method is applied, as it involves the subjective decision on how to model the problem.

Second limitation of this work is taking into account only vulnerabilities related to business assets. It influences on the whole security level of an enterprise. The omission of developing risk-related MAD for IS assets makes system vulnerable even if all business assets related risks are mitigated.

Talking about the used example, it is focus on specific attack methods (*e.g.*, keylogger and SQL injection). However many other attacks could be also used in order to violate the system (*e.g.*, man in the middle attack *etc.*). Although the example is taken from study case, which is based on real world example, the current approach was no applied on the real EA model.

7.2 Conclusions

7.2.1 *The Institut Luxembourgeois de la Normalisation, de l'Accréditation*

Countermeasures that were implemented in illustrated example are taken from our own experience and are not based on any regulation document. However it would be nice to compare implemented controls with countermeasures proposed in regulation documents. Regulation documents provide enterprise engineers with information about control implementation for security maintenance. Among variety of available regulation documents (*e.g.*, Basel II [4], The Institut Luxembourgeois de la Normalisation, de l'Accréditation (ILNAS) [22], Sarbanes-Oxley Act [49] and Directive 2009/140/EC [41]), the scope of requirements, which are identified in ILNAS, is the most suitable for our example. ILNAS discusses information security management system and operational management system.

To verify the correctness of implemented controls, it is needed to check them accordingly ILNAS based criteria list. This list should be done by analyst who will apply the current method. He/she should read through ILNAS regulation document and verify if the

implemented controls are mentioned there. The main difficulty is the difference in terminology. Talking about illustrated example, the misuse of authority to get access to the system is discussed in “Security rights profile” chapter of ILNAS. Proposed in example logging alerts are mentioned in chapter “Monitoring mechanisms to identify logs”. The email filtering, which was defined in illustrated example could be determined through chapter “Mechanism for found and eliminating malicious code in the digital documents collected for archiving”.

7.2.2 Answer to RQ

The research question was identified in Chapter 1 and it sounds as “*How to align Enterprise Architecture and Security Risk Management?*”. To answer this question, firstly, we have investigated different RM concepts and approaches. After analysis and comparison ISSRM DM was chosen as SRM method. For visual presentation of risk-oriented problem, MAD modeling language was chosen. ArchiMate is one of the approaches that could be used to build EA model. To align SRM and EA it is needed to make a mapping between modeling languages that are used to present these concepts. Hence, the alignment between MAD and ArchiMate should be done.

7.3 Future work

For future work it is necessary to negate all listed limitations that appeared in this paper. The main work that should be done is constructing transformation rules and extension of current approach for IS assets vulnerabilities. Actually, the sequence of steps and main rules will remain unchanged. The only thing that needs to be implemented in order to extend the current approach for mitigation of IS assets-related risks is transformation back rules from risk treatment-related concept to ArchiMate model.

It is necessary to come up with same terminology for ILNAS and ArchiMate. Since verification of controls, implemented into the system, should be checked through regulatory documents, the criteria list should be identified based on ILNAS and analyst should just check if the countermeasures satisfy these criteria.

The alignment between EA and SRM was based on two modeling languages (ArchiMate and MAD). However there are more languages for SRM presentation and each of them has it's own perspective. The implementation of alignment of ArchiMate to other risk-oriented modeling languages (Secure Tropos, Misuse cases, BPMN *etc.*) will help to choose the most suitable alignment depending on the need of preferences of the analyst.

8 References

- [1] Alberts C. J. and Dorofee A. J., "OCTAVE criteria, Version 2.0," Carnegie Mellon University - Software Engineering Institute, Pittsburgh, Pennsylvania, CMU/SEI-2001-TR-016, 2001.
- [2] Altuhhova O., Matulevicius R., Ahmed N., "An Extension of Business Process Model and Notation for Security Risk Management", International journal of Information system Modelling and Design (IJISMD), 2013.
- [3] AS/NZS 4360, "Risk management", SAI Global, 2004.
- [4] Basel Committee on Banking Supervision, "Sound Practices for the Management and Supervision of Operational Risk", Feb-2003.
- [5] Braber F. D., Hogganvik I., Lund M. S., Stølen K. and Vraalsen F., "Model-based security analysis in seven steps—a guided tour to the CORAS method.", BT Technology Journal, Volume 25 Issue 1, pages 101–117, 2007.
- [6] Bresciani P., Perini A., Giorgini P., Fausto G. and Mylopoulos J., "TROPOS: an Agentoriented Software Development Methodology", Journal of Autonomous Agents and Multi-Agent Systems, Volume 25, pages 203–236, 2004.
- [7] Bundesamt für Sicherheit in der Informationstechnik, "The IT-Grundschutz Catalogues". 2005.
- [8] Chowdhury M. J. M., Matulevičius R., Sindre G., Karpati P., "Aligning Mal-activity Diagrams and Security Risk Management for Security Requirements Definitions", 2012.
- [9] CLUSIF, MEHARI, "Concepts and Mechanisms" France, 2007.
- [10] Common Criteria. (2006) Common Criteria for Information Technology Security Evaluation, Version 3.1. [Online]. <http://www.commoncriteriaportal.org/files/ccfiles/CCPART1V3.1R1.pdf>
- [11] DCSSI, "Section 1 - Introduction. EBIOS - Expression of Needs and Identification of Security Objectives", France, 2004.
- [12] Dubois, E., Heymans, P., Mayer, N. & Matulevičius, R., "A Systematic Approach to Define the Domain of Information System Security Risk Management", In: S. Nurcan, C. Salinesi, C. Souveyet & J. Ralyte, eds. *Intentional Perspectives on Information Systems Engineering*. s.l.: Springer-Verlag, pp. 289-306, 2010.
- [13] EBIOS, "Etude de cas @rchimed", France, July 2004.
- [14] EBIOS. (2004) Expression of Needs and Identification of Security Objectives. [Online]. http://www.ssi.gouv.fr/en/con_dence/ebiospresentation.html
- [15] G. Stoneburner, C. Hayden, and A. Feringa, "NIST Special Publication 800-27 Rev. A: Engineering Principles for Information Technology Security (A Baseline for Achieving Security)", Gaithersburg: National Institute of Standards and Technology, 2004.
- [16] Gandhi R. , Lee S.W., "Ontology Guided Risk Analysis:From Informal Specificatopns to Formal Metrics", *Advances in Information & Intelligent Sys.*, 2009

- [17] Giorgini P., Mouratidis H., Zannone N., "Modeling Security and Trust with Secure Tropos", Information Science Publishing, p.161, 2007.
- [18] Grandry E., Feltus C., Dubois E., "Conceptual Integration of Enterprise Architecture Management and Security Risk Management", Enterprise Distributed Object Computing Conference Workshops (EDOCW), 17th IEEE International, 2013.
- [19] Herold S., Klus H., Welsch Y., Rausch A., Reussner R., Krogmann K., Koziolok H., Mirandola R., Hummel B., Meisinger M., Pfaller C., "CoCoME – The Common Component Modeling Example", The Common Component Modeling Example, pages 16-53, 2008
- [20] IFIP-IFAC Task Force. (March, 1999) GERAM: Generalised Enterprise Reference Architecture and Methodology. [Online]. <http://www.ict.griffith.edu.au/~bernus/taskforce/geram/versions/geram1-6-3/GERAMv1.6.3.pdf>
- [21] Information Security Management Systems (ISMS), BSI Standard 100-1, Version 1.5, May 2008
- [22] ILNAS, "Technical regulation requirements and measures for certifying Digitisation or Archiving Service Providers (PSDC)", Feb-2013.
- [23] Innerhofer-Oberperfler F. and Breu R., "Using an enterprise architecture for IT risk management", ISSA, 2006.
- [24] Insight Consulting, "CRAMM (CCTA Risk Analysis and Management Method) User Guide version 5.0.", SIEMENS, 2003.
- [25] ISO/IEC 13335-1, "Information technology – Security techniques – Management of information and communications technology security – Part 1: Concepts and models for information and communications technology security management", Geneva: International Organization for Standardization, 2004.
- [26] ISO/IEC 27001, "Information technology – Security techniques – Information security management systems – Requirements", Geneva: International Organization for Standardization, 2005.
- [27] ISO/IEC 27005, "Information technology – Security techniques – Information security risk management", Geneva: International Organization for Standardization, 2008.
- [28] ISO/IEC, "Guide 73, Risk management – Vocabulary – Guidelines for use in standards", Geneva: International Organization for Standardization, 2002.
- [29] Jurjens J., "UMLsec: Extending UML for Secure Systems Development", 5th International Conference Dresden, Germany, September 30 – October 4, 2002.
- [30] Lamsweerde A. V., "Elaborating Security Requirements by Construction of Intentional Antimodels", In the proceedings of the 26th International Conference on Software Engineering, 2004.
- [31] Lee S. W., Gandhi R. A., and Wagle S., "Towards a Requirements-driven Workbench for Supporting Software Certification and Accreditation", In proceeding of the 3rd International Workshop on Software Engineering for Secure Systems, 2007.

- [32] Lee S. W., Gandhi R., Muthurajan D., Yavagal D. and Ahn G. J., "Building problem domain ontology from security requirements in regulatory documents", In proceeding of the International Workshop on Software Engineering for Secure Systems, 2006.
- [33] Loddeerstedt T., basin D., Doser J., "SecureUML: A UML-Based Modeling Language for Model-Driven Security", Proceedings of the 5th International Conference on The Unified Modeling Language, 2002.
- [34] Matulevičius R., Mouratidis H., Mayer N., Dubois E., "Syntactic and Semantic Extensions to secure Tropos to Support Security Risk Management", 2012.
- [35] Matulevičius R., Lakk H., Lepmets M., "An approach to assess and compare quality of security models", ComSIS, Volume 8 No 2, Special Issue, (2011), pp.447-476, 2011.
- [36] Matulevičius R., Mayer N. and Heymans P., "Alignment of Misuse cases with Security Risk Management", In Proceedings of the 3rd International Conference on Availability, Reliability and Security, 2008.
- [37] Matulevičius R., Mayer, N., Heymans, P., "Adapting Secure Tropos for Security Risk Management in the Early Phases of Information Systems Development", In: CAiSE '08 Proceedings of the 20th international conference on Advanced Information Systems Engineering, pp.541-555, Springer-Verlag, 2008
- [38] Mayer N., "Model-Based Management of Information System Security Risk", Doctoral Thesis in Computer Science Namur, Belgium, 2009.
- [39] McDermott J. and Fox C., "Using Abuse Case Models for Security Requirements Analysis", In Proceedings of the 15th Annual Computer Security Applications Conference, 1999.
- [40] Mouratidis H. and Giorgini P., "Secure Tropos: A Security-oriented Extension of the. Tropos Methodology", International Journal of Software Engineering and Knowledge Engineering, 2005.
- [41] Official Journal of the European Union, "Directive 2009/140/EC of the European Parliament and of the Council of 25 November 2009", 2009.
- [42] Remco, M., Dijkman, R.M., Dumas, M., & Ouyang, C. (2007). Formal Semantics and Analysis of BPMN Process Models using Petri Nets. In Journal Information and Software Technology. Elseiver.
- [43] Sindre G. and Opdahl A. L., "Capturing security requirements by misuse cases", In proceedings of the 14th Norwegian informatics conference, 2001.
- [44] Sindre G., "Mal-Activity Diagrams for Capturing Attacks on Business Processes", In proceedings of the Working Conference on Requirements Engineering: Foundation for Software Quality, 2007.
- [45] Sindre G., Opdahl A. L., "Eliciting Security Requirements with Misuse Cases", Requirements Engineering, 10(1), pp.34-44, 2005.
- [46] Soomro I., Ahmed N., "Towards Security Risk-oriented Misuse Cases", International Workshops, Tallinn, Estonia, September 3, 2012.
- [47] Stoneburner G., Hayden C., and Feringa A., "NIST Special Publication 800-27 Rev. A: Engineering Principles for Information Technology Security (A Baseline

- for Achieving Security)", Gaithersburg: National Institute of Standards and Technology, 2004.
- [48] The Open Group. (2013) "ArchiMate 2.1 Specification". [Online]. <http://pubs.opengroup.org/architecture/archimate2-doc/toc.html>
- [49] U. S. Senate and H. of R. in C., "Sarbanes-Oxley Act of 2002", 2002.
- [50] Vraalsen F., Mahler T., Lund M. S., Hogganvik I., F. den Braber, and Stølen K., "Assessing Enterprise Risk Level: The CORAS Approach," in Advances in Enterprise Information Technology Security, D. Khadraoui and F. Herrmann, Eds. Idea group, pp.311–333, 2007.
- [51] Zachman J. (2011) The Zachman Framework Evolution. [Online]. <http://www.zachman.com/ea-articles-reference/54-the-zachman-framework-evolution>


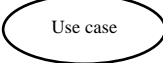





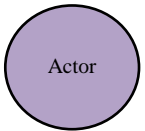


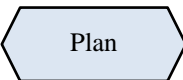

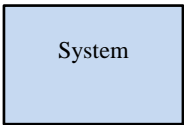
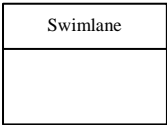
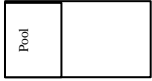





Appendix

I. Alignment of modeling languages with ISSRM DM

All four modeling languages should be observed regarding to the alignment with ISSRM DM. The observation will be based on the comparison according to the ISSRM DM constructs.

The main terms of asset-related construct of ISSRM DM and the correspondence of analyzed languages are presented in Table I.1. The term *asset* in general stands for anything that has been valued for the organization and it is necessary for achieving its goals. Assets could be divided into IS and business assets [12]. Comparing the languages we can see that only BPMN has separated semantic for each term. Although misuse cases and mal-activity diagram also have specific syntax for IS asset, (e.g., misuse cases use system scope to determine IS assets and mal-activity diagrams use swimlane) they still do not specify business asset. Secure Tropos does not have at all separation of assets. It presents assets as combination of actor, hardgoal, resource and plan. The connection is made using dependency, contribution, means-ends, and decomposition links.

Table I.1. Language correspondence to ISSRM regarding asset-related concept

ISSRM	Misuse cases	Mal-activity diagrams	BPMN	Secure Tropos
Asset	 Actor  Use case combined using extends and includes links	 Decision  Activity combined using control flow links	 Task  Event  Gateway combined using Sequence flows	 Actor  Hardgoal  Resource  Plan
Business asset	combined using extends and includes links		 Data Object	
IS asset	 System	 Swimlane	 Pool  Data Store	Combined together using dependency, contribution, means-ends, and decomposition links
Security criterion	 Security criterion	-	 Added to the business asset	 Security constraint  Softgoal combined using contribution and security constraint decomposition links

Security criterion is a property or constraint on business assets that characterizes their security needs [12]. In case of misuse cases and BPMN, it specifies exactly the place where security criterion is needed. Mal-activity diagrams do not specific syntax for security criterion at all, what makes them less understandable than misuse cases. Secure Tropos defines security criterion using security constraint and softgoal.

In Table I.2. the aforementioned languages correspond to the ISSRM DM regarding to risk-related concept. *Risk* is a combination of threat with one or more vulnerabilities leading to a negative impact harming one or more of the assets [12]. There is no syntax in any of analyzed languages that could allow expressing risk in one construction. In all of aforementioned languages risk could be defined as combination of event and impact constructions.

Impact is a potential negative consequence of the risk that may harm assets when the threat is accomplished [12]. Misuse cases and mal-activity diagrams syntax gives us full picture of impact of the risk occurrence. In misuse case it is presented using the impact construct. In mal-activity diagrams it is done through mal-activity construct, which is defined in the thread, expressed swimlane. In case of BPMN and Secure Tropos provision of the impact syntax is not so clear. BPMN specifies only harm of business asset, but not IS asset. Although Secure Tropos shows the negation of security criterion, it does not show the complete impact on the assets.

Event is a combination of a threat and one or more vulnerabilities [12]. In all provided modeling languages there is no speared construction to express event. It could be done as a combination of threat and vulnerability. However Secure Tropos has additional construct that allow presenting this term. It is threat construct, apart from previously defined combination.

Vulnerability is a characteristic of an IS asset that can contribute a weakness or a flaw in terms of IS security [12]. Misuses cases give the whole picture of vulnerabilities of the system. In mal-activity diagrams this term does not have at all separate syntax. In case of BPMN and Secure Tropos only vulnerability point could be identified.

Threat is a potential attack, carried out by an agent that targets one or more IS assets and may lead to harm to assets [12]. In misuse case, mal-activity diagrams and BPMN there is not construct that defines threat. It could be done through combination of attack method and threat agent. However Secure Tropos determines threat combining hardgoal and plan, which are specified in treat agent construct.

Attack method is a standard means by which a threat agent carries out a threat [12]. All languages provide full definition of attack method. In each case it is done by combination of several modeling constructs. Misuse cases provide it through misuse case construct, which is combined with other misuse cases using links. The same is done in Secure Tropos, but instead misuse case it have plan construction. Mal-activity and BPMN diagrams have more complicated syntax.

Threat agent is an agent that can potentially cause harm to the assets of the IS [12]. In all languages there present clear construct, which identifies threat agent. Misuse cases specify it as misuser. Mal-activity diagrams show it as a mal-swimlane. BPMN determines threat agent as a pool. Secure Tropos provide information about threat agent through an actor construct.

Table I.3 provides information about correspondences between languages and risk treatment-related concepts. *Risk treatment* is a decision of how to treat the identified risk [12]. Since it is difficult to show this decision in visual concept, there is no construct determining it.

Security requirement is a condition over the phenomenon of the environment that should come true by installing the IS in order to mitigate risks [12]. The same as in specification of assets, it is done through combination of different constructs.

Control is a designed means to improve security, specified by a security requirement and implemented to comply it [12]. There are no constructs provided by misuse cases, BPMN and Secure Tropos. Only mal-activity diagrams could show control, which could be implemented through usage a swimlane, which will contain mitigation activities.

Table I.2. Language correspondence to ISSRM regarding risk-related concept

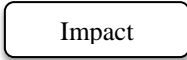
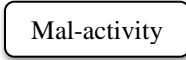









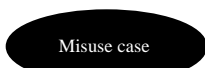

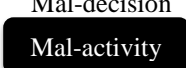
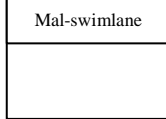



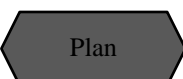

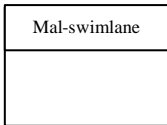
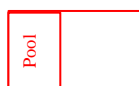







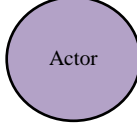
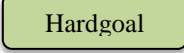

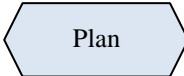
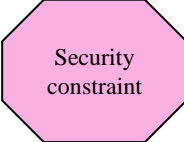
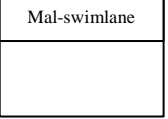
ISSRM	Misuse cases	Mal-activity diagrams	BPMN	Secure Tropos
Risk	Combination of Event and Impact			
Impact		 Contained in the malswimlane that expresses attack method		 Impacts
Event	Combination of Vulnerability and Threat			 Or combination of Vulnerability and Threat
Vulnerability		-	 added to the IS asset	 added to the IS asset
Threat	Combination of Attack method and Threat agent			  
Attack method	 combined using includes and extends	  combined using control flow Presented in 	   Event Gateway combined using Sequence flows	 combined with other Tasks using decomposition links
Threat agent	 Misuser			 Actor

Table I.3. Language correspondence to ISSRM regarding risk treatment-related concept

ISSRM	Misuse cases	Mal-activity diagrams	BPMN	Secure Tropos
Risk treatment	-			
Security requirement	 <p>combined using extends and includes links</p>	 <p>decision</p>  <p>Mitigation activity</p> <p>combined using control flow</p>	 <p>Task</p>   <p>Event Gateway</p> <p>combined using Sequence flows</p>	 <p>Actor</p>  <p>Hardgoal</p>  <p>Resource</p>  <p>Plan</p>  <p>Security constraint</p> <p>combined using dependency, contribution, means-ends, and decomposition links</p>
Control	-	 <p>Mal-swimlane</p>	-	-

II. License

Non-exclusive licence to reproduce thesis and make thesis public

I, **Iuliia Tovstukha** (date of birth: 10.09.1991),

(Iuliia Tovstukha)

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

Management of Security Risks in the Enterprise Architecture using ArchiMate and Mal-activities,

(Management of Security Risks in the Enterprise Architecture using ArchiMate and Mal-activities)

supervised by Raimundas Matulevičius,

(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, **26.05.2014**