UNIVERSITY OF TARTU
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

Institute of Computer Science
Software Engineering Curriculum

Servet Kurt

Interplay of Misuse Case and Fault Tree Analysis for Security and Safety Analysis

Master's thesis (30 ECTS)

Supervisor: Dr. Raimundas Matulevičius

Tartu 2014

Interplay of Misuse Case and Fault Tree Analysis for Security and Safety Analysis

Abstract

Nowadays safety and security are becoming more and more important because of the fact that modern information systems are increasingly distributed over web-services, grids and clouds. Safety critical systems that were not utilizing usage over Internet are being re-engineered in order to be use over Internet. As a consequence of this situation there is need of new methods that cover both security and safety aspects of software systems, since these systems are used in transportation, health and process control systems that arises risk of physical injury or environmental damage. Additionally when safety and security aspects are not considered together they may violate each other while one situation is making a case safe it may violate security and this is a problem. Such as in the sample of lock doors at dormitories for security purpose to protect inhabitants against robbery and some other possible crimes, those inhabitants of dormitories use distance keys to unlock them but in case of a fire situation in the building for safety purposes these lock doors are unlocking themselves and by activating fire alarms attackers can get access to inhabitants properties. In current thesis we introduce integrated domain models of security and safety, extracting definitions from safety and security domains and finding possible pairs to integrate. Developing interplays between security and safety technique that is misuse cases and fault tree analysis. We demonstrate alignment of fault tree analysis to safety domain model and making interplay between techniques from fault tree analysis to misuse cases. By using the domain models of both security and safety and making interplay between techniques we proposed an integrated technique we expect to solve the problem to cover both safety aspects of software system benefiting from complementary strengths of security domain model and techniques.

We believe that our study is contributing to the integration attempts of security and safety techniques by illustrating alignment of fault tree analysis with safety domain model benefitting from misuse cases and information security risk management relationship and making interplay with misuse case technique. And also we illustrate a new methodology on how to use fault tree analysis and misuse cases in order to elicit safety concerns in a new information system by having interplay with misuse case. Moreover, we test correctness of our methodology by making results comparison of a safety risk analyze done.

Key Words:

Safety Domain Model, Fault Tree Analysis, Misuse Cases, Security Risk Management.

Misuse case´i ja Fault Tree Analyse´i koosmõju turvalisuse ja ohutuse uurimisel

Sisukokkuvõte

Ohutus ja turvalisus infosüsteemides muutuvad aasta-aastalt üha olulisemaks. Seda seetõttu, et kaasaegsed infosüsteemid on üha enam levinud veebiteenustes, -võrgustikes ja – pilvedes. Ohutuse seisukohalt olulisi süsteeme, mida ei ole varem Internetis kasutatud, tehakse ümber, et muuta neid kasuatatvaks Internetis. Selle tulemusena on tekkinud vajadus leida uusi meetodeid, mis kindlustaks nii ohutuse kui turvalisuse tarkvarasüsteemides. Kui ohutust ja turvalisust ei käsitleta koos, võivad nad riske suurendada – olukorra ohutuks muutmine võib tekitada riski turvalisuses ning sellest tekib probleem. Näiteks lukustatud uksed ühiselamutes turvalisuse huvides, kaitsmaks sealseid elanikke röövide ning muude võimalike kuritegude eest. Uste avamiseks kasutavad ühiselamu elanikud kaarte, mis uksed avavad. Tulekahju korral aga avanevad uksed ohutuse eesmärgil automaatselt ning kurjategijad, lülitades sisse tuletõrjealarmi, pääsevad ühiselamu elanike vara juurde.Antud uurimistöös antakse ülevaade ohutusest ja turvalisusest kui ühtsest süsteemist, määratledes ohutuse ja turvalisuse mõisted ning otsides võimalikke viise nende integreerimiseks, arendades koosmõju ohutuse ja turvalisuse vahel kasutades misuse case´i ja fault tree analysis´i. Töös selgitatakse fault tree analysis´i sobivust ohutuse domeeni mudelisse ja püütakse leida koosmõju fault tree analysis´i ja misuse case´i tehnikate vahel.

Kasutades nii ohutuse kui turvalisuse domeenimudeleid ning tekitades koosmõju tehnikate vahel, on oodatud tulemuseks ohutuse ja turvalisuse probleemi lahendamine tarkvarasüsteemides. Usutavasti aitab antud uurimistöö kaasa ohutuse ja turvalisuse integreerimisvõimaluste leidmisele selgitades fault tree analysis sobivust ohutuse domeenimudelisse, kasutades misuse case´i ja information security risk management´i seost ja kooskõlastades seda misuse case´i tehnikaga Samuti selgitatakse töös uut metoodikat, kuidas kasutada fault tree analysis-d ja misuse case´i selleks, et saavutada nii ohutus kui turvalisus kaasaegsetes infosüsteemides. Lisaks sellele testiti töös selgitatud sobivust usaldusväärse stsenaariumi korral, mis kinnitab sobivuse paikapidavust.

Võtmesõnad:

Ohutuse domeenimudel, fault tree analysis, misuse cases, turvalisuse riskide haldamine

Acknowledgement

In the first place, I would like to express my deepest appreciation to my supervisor Dr. Raimundas Matulevičius, who has been a tremendous mentor for me. Definitely, his guidance was the key factor for me to understand the vision of this thesis.

I offer my regards and blessing to my brother Berkan Kurt, who has been like a hand for me in Turkey with my spoiled demands. Thanks for your patience.

I would also want to emphasize my deep gratitude to my beloved fiancé Kadri Heina, for helping me to keep my concentration on my studies with her constant support, encouragement and love.

Most importantly, none of this would have been possible without the love and patience of my parents Ahmet Kurt and Serpil Kurt, who supported me both mentally and financially all these years up to this level.

Servet Kurt

Table of Contents

Table of Figures

List of Tables

CHAPTER 1. Introduction

Trustworthiness of an information system establishes its dependability that allows reliance to be justifiable placed on the service it provides. Dependability requirements are the process of elicitation and validation of security and safety requirements of a new information system. There are different techniques for elicitation and validation of dependable requirements. All these techniques have different rules, methodologies and elements. However, since these technologies are all concentrated on dependable requirements and all these dependable requirements concentrate on what an IS should not do they have similarities. Although there is an express difference between the philosophies of security and safety as intentional versus accidental, techniques for identifying hazards and threats are common in principle. We know that these techniques for security and safety requirements elicitation and validation are similar and complementary to each other. Therefore can be used together to generate interplay between techniques.

We foresee the possibility of integrated technique of security and safety. Since security and safety techniques have cross-pollinations, similarities and each one of these have weaknesses and strengths such as safety techniques are more standardized and security techniques are more visual. And most important of all they both focused on what an IS system should not do. Also when it is considered in some researches security techniques are used as safety techniques by doing some conversions in their structures (Winther et al., 2001). But yet preliminary investigations showed that there is no actual usage of integration for security and safety techniques. Therefore we believe that our study will be make contribution to integration attempts of security and safety.

In this thesis information security risk management (ISSRM) and safety domain model (SDM) three security risk management languages; Secure Tropos, Mal-Activity Diagrams (MAD), Misuse Cases, four safety risk management languages; Hazard and Operability (HAZOP), Preliminary Hazard Analysis (PHA), Fault Tree Analysis (FTA), Failure Mode and Effect Analysis (FMEA) are covered. Also Misuse Cases ant its alignment to ISSRM (Soomro, 2012) covered and FTA and its alignment to SDM illustrated.

In this thesis we resolve two research questions.

RQ1: What are the similarities and differences between security and safety domain models and how can they be benefitted from each other?

RQ2: What kind of interplay can be done between security and safety techniques for security and safety analysis?

Our contribution in this thesis is FTA and its alignment to SDM. In order to do this alignment we have illustrated methodology of FTA along with its ground rules and tested it with SDM on running example. From this test we discovered that FTA could only analyze risks related concepts from SDM. Therefore, we have decided to use Misuse Case to cover asset model and safety requirement model from SDM. By making interplay between techniques we could align FTA with SDM. We have also provided step-by-step procedure of how to apply FTA and its alignment to SDM in our contribution.

The thesis consists of six chapters. Chapter 1 presents the overview of thesis along with motivation, scope, research question, contribution and structure. Chapter 2 illustrates security and safety domain models by giving concept definitions; security domain model concept definitions extracted from (Mayer, 2009) and (Dubois et al., 2010) and definitions related to SDM composed from (Firesmith, 2003). And presents similarities and differences between security and safety domain models. In chapter 3, modelling languages for security and safety risk management are given. These are Secure Tropos (Matulevičius et al., 2012), MAD (Sindre, 2007), Misuse Case (Sindre et al., 2004) from security and HAZOP (Kletz, 1999), PHA (Ericson, 2005), FMEA (Stamatis, 1995), FTA (Brooke et al., 2003) from safety. Chapter 4 presents contribution that is FTA and its alignment to SDM showing cross-pollinations between techniques Misuse Case and FTA along with methodology and application of the alignment. Chapter 5 describes the validation of the contribution in terms of its correctness. Finally, Chapter 6 provides the conclusion and future work suggestions.

CHAPTER 2. Security and Safety Domains

This chapter introduces security and safety domain models along with their definitions and samples to concepts. It also gives similarities and differences of security and security domain models.

2.1 Security Domain Model

Security domain model is the theoretical or conceptual model of all security related risks and its various relationships, constraints, attributes to prevent or reduce the severity of the risk impact on asset. Main focus of security domain model is security risk management that is describing the road map of identifying valuable assets, deciding on risk levels and risk treatments, dictating security criterion for reducing the likelihood of undesirable events. There exist several security risk management methodologies as CORAS (Braber et al., 2007), Automated Risk and Utility Management (AURUM) (Ekelhart et al., 2009), ISSRM (Dubois et al., 2010) and Goal-Risk driven assessment (Asnar et al., 2010). In our study we chose to use ISSRM as our security domain model becauese it presents three major groups we think it is important in risk identification as asset-related concepts, risk-related concepts and risk treatment-related concepts (Mayer et.al., 2009). Moreover, ISSRM is applicable to Misuse-Cases that is the security risk identification technique and going to be used to make interplay with (Soomro, 2012).

2.1.1 ISSRM Definitions

The definitions related to ISSRM study is done by (Mayer, 2009) and (Dubois et al., 2010). In Mayer's study extracted definitions from different resources are collected. This paper will introduce definitions for concepts of ISSRM Domain Model presenting them from Dubois's definitions and selecting them through Mayer's collection of definitions.

The definitions related to ISSRM are extracted from (Dubois et al., 2010)

Asset: Anything that has value to the organization and necessary for achieving its objectives.

Business asset: Information, process, skill inherent to the business of the organization that has value to the organization in terms of its business model and is necessary for achieving its objectives.

IS asset: A component or part of the IS that has value to the organization and is necessary for achieving its objectives and supporting business assets.

Security criterion: Property or constraint on business assets that characterizes their security needs. Security criteria acts as indicators to assess the significance of a risk.

Risk: The combination of a threat with one or more vulnerabilities leading to a negative impact harming one or more of the assets. Threat and vulnerabilities are part of the risk event and impact is the consequence of the risk.

Impact: The potential negative consequence of a risk that may harm assets of a system or an organization, when a threat (or an event) is accomplished.

Event: The combination of a threat and one or more vulnerabilities.

Vulnerability: The characteristic of an IS asset or group of IS assets that can constitute a weakness or a flaw in terms of IS security.

Threat: Potential attack, carried out by an agent that targets one or more IS assets and that may lead to harm to assets. A threat is constituted of a threat agent and an attack method.

Threat agent: An agent that can potentially cause harm to assets of the IS. A threat agent triggers a threat and is thus the source of a risk.

Attack method: Standard means by which a threat agent carries out a threat.

Risk treatment: The decision of how to treat the identified risks. A treatment satisfies a security need, expressed in generic and functional terms, and can lead to security requirements.

Security requirement: A condition over the phenomena of the environment that we wish to make true by installing the IS, in order to mitigate risks.

Control (also called countermeasure or safeguard): A designed means to improve security, specified by a security requirement, and implemented to comply with it.

2.1.2 Samples to Security Domain Model Concepts

Security domain model has some concepts as it has shown in *Fig 1*. In our example we will focus on giving samples to each of these concepts regarding a car theft scenario.
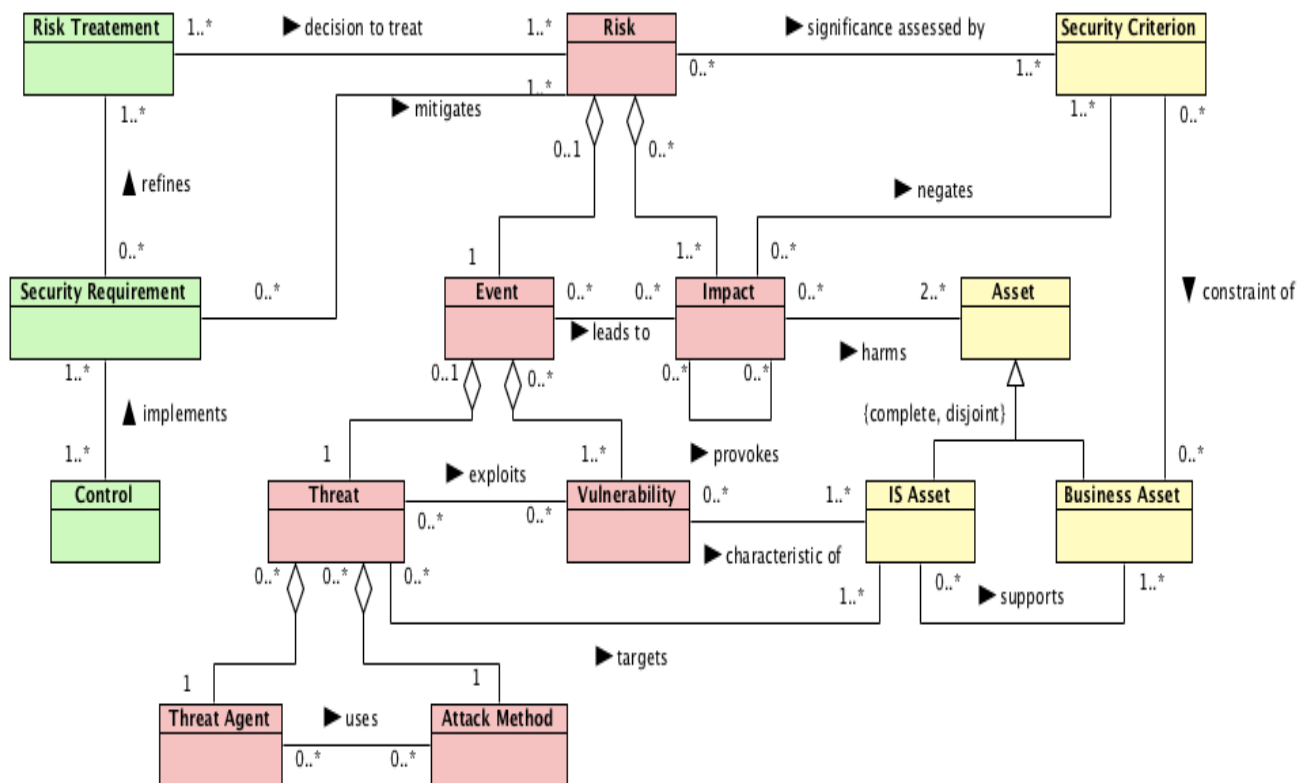
Fig 1. The ISSRM Domain Model; adopted from (Dubois et al., 2010) and (Mayer, 2009)

A medium sized telecommunication company has company cars for maintenance works. Employee is using these cars. None of these cars have immobilizer system. Car thieves having the knowledge of which car has immobilizer which not and how to hot wire a car targeting this company's car for stealing purpose.

Table 1 illustrates samples to each class of the domain model in Fig 1 regarding the scenario above.

<div align="center">Table 1. Samples to Security Domain Model Concepts</div>

Asset	Company property.
Business asset	Car.
IS asset	Ignition start.
Security criterion	Confidentiality of the car.
Risk	Car thief steals the car.
Impact	-Negates the integrity of the car.
Event	Car thief uses screwdriver to accomplish hot wire technique in order to start the engine and since the car has no immobilizer system avoiding it being hot-wired car thief starts the engine successfully and steals the car.
Vulnerability	Unsecure ignition start, no immobilizer system avoiding the car being hot-wired.
Threat	Car thief using hot wire technique to steal the car.
Threat agent	Car thief using screwdriver.
Attack method	-Insert a slotted screwdriver into the ignition and turn over it like a regular key.
Risk treatment	Risk avoidance.
Security requirement	Car engine should not be able to start unless the correct key (or other token) used.
Control	Correct key triggers a small electromagnetic field, which induces current to flow inside the key body, which in turn broadcasts a unique binary code that is read by the automobile's Engine Control Unit (ECU). When the ECU determines that the coded key is both current and valid, the ECU activates the fuel-injection sequence and the engine starts.

2.2 Safety Domain Model

SDM is the theoretical or conceptual model of all safety related risks and its various relationships, constraints, attributes to prevent or reduce the severity of the risk harm on asset. Comparing to safety, security is more embedded to guidelines and standards however yet our

preliminary investigations showed there exist no domain model for safety. As a consequence of this adopting the information model for safety engineering presented by (Firesmith, 2003) we have proposed SDM that is shown in Fig 2.

2.2.1 Definitions Related to SDM

The definitions related to SDM composed by us from (Firesmith, 2003). We demonstrate definitions for all concepts of SDM Fig 2.

Asset: Asset is everything that could be considered as valuable to protect from harm. The why reason asset needs protection is because it affects from accident somehow. Such as being part of the accidental system, external to the accidental system, future participant of the accidental system or participants having no intentional involvement to the system.

Safety goal: Safety goal is the target of safety level or one of safety sub factors that is expected to meet by the information system. These safety sub factors are asset protection, safety incident detection, safety incident reaction and system adaptation.

Safety risk: Safety risk is accidental harm to an asset that is possible to occur when potential risks come true due to ignoring actual safety risk or vulnerability.

Accident: Accident is series of events or one single event resulting with harm to an asset that is unplanned and unintended of creating it. Accidents may result in damage to environment, health or property.

Vulnerability: Vulnerability is weak points in the system in terms of safety that is increasing the chance of possible accidental situation. These weak points occur due to design, implementation, architecture or deployment mistakes and result with harm to the asset.

Hazard: Hazard is the situations that is potentially or actually increasing the chances of accidental situations. Hazardous states can be both potential and actual.

Harm: Harm is specific damage an asset gets because of a hazardous situation turning into reality and causing accidental situation.

Safety policy: Safety policy is defined unacceptable potential risks by software engineers. It mandates information system to follow the safety policy context in order to avoid situations resulting with harm.

Safety requirement: Safety requirement specifies and dictates the minimum needed measure of safety information system has to reach to ensure quality in terms of safety.

Safety mechanism: Safety mechanism is the architectural mechanism that aims to reduce chances of accidental situations by fulfilling safety requirements.
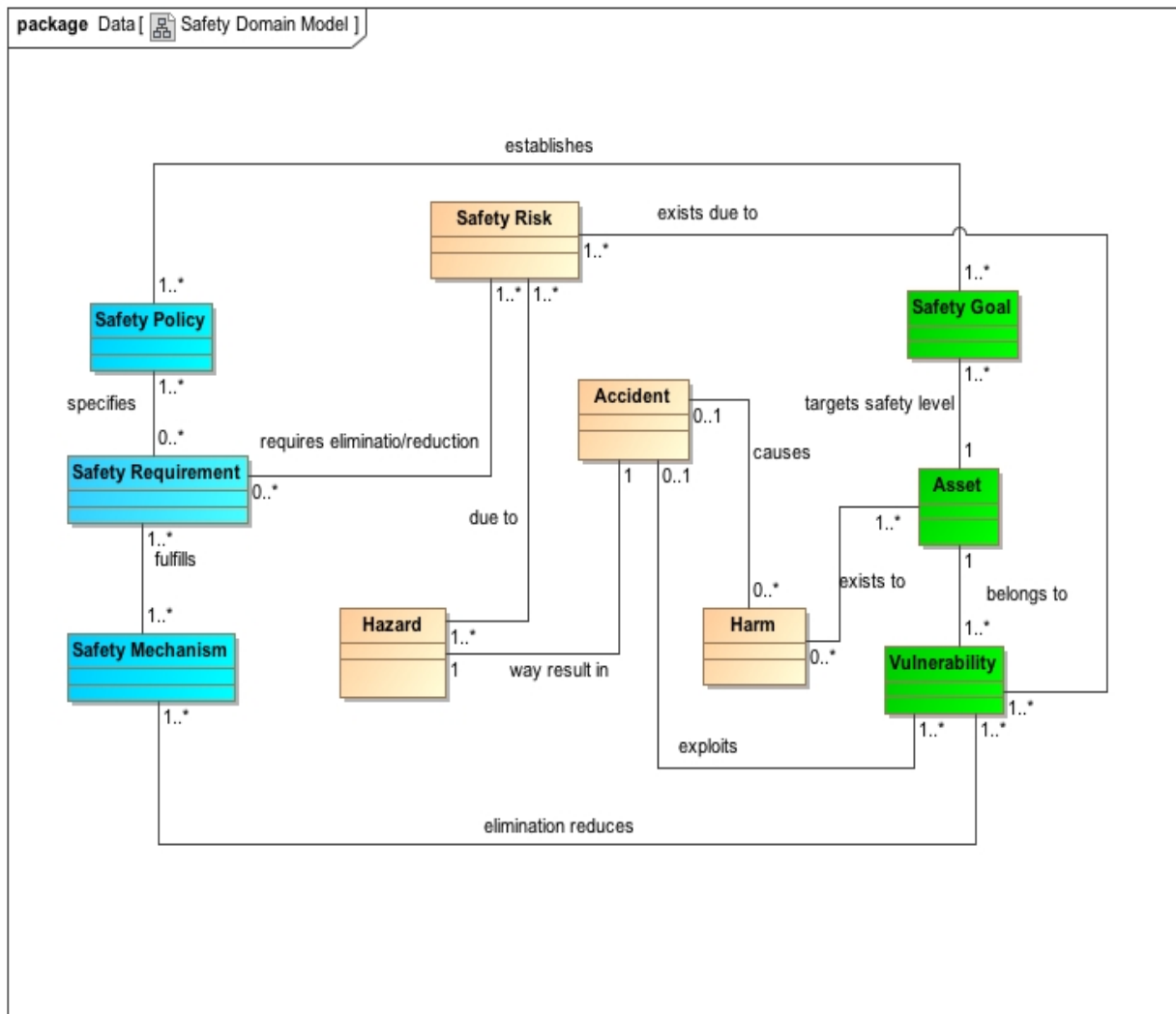
Fig 2. SDM; adopted from (Firesmith, 2003)

2.2.2 Samples to SDM Concepts

SDM has concepts as it has shown in Fig 2. In our sample we will focus on giving examples to each of these concepts within the context of the following scenario. At a dormitory due to privacy concerns there exists lock doors at each floor for security purposes to protect inhabitants against robbery and some other possible crimes, those inhabitants of dormitories use distance keys to unlock them to get in to their floors and get out. In case of a fire situation these doors are getting unlocked automatically. However there exist no manual way of opening these doors to put it another way if the doors would not open automatically in fire situation there is no manual way to open them. Thus in a fire situation dormitory inhabitant may not vacate the premises and got poisoned by smoke.

Table 2 illustrates samples to each concept of the domain model in Fig 2 regarding the scenario above.

Table 2. Samples to SDM Concepts

Asset	Dormitory inhabitant.
Safety goal	Asset protection.
Safety risk	Dormitory inhabitant cannot vacate the premises.
Accident	Dormitory inhabitants get stuck in the building.
Vulnerability	There exist no way to unlock the doors manually.
Hazard	Lock doors not get unlocked in fire situation.
Harm	Dormitory inhabitant that got poisoned by smoke
Safety policy	When the fire alarm is active lock doors should automatically get unlocked.
Safety requirement	In order dormitory inhabitants to vacate the premises as soon as possible all lock doors should be unlocked.
Safety mechanism	Smoke detectors should activate the fire alarm and extinguishers. Activated fire alarm should unlock all the lock doors. In order lock doors to open manually lock covers should be made of plastic thus inhabitants can break them and vacate the premise

2.3 Correspondence between Security and Safety Domains

We believe that domain models of safety and security can give an idea on how to make interplay between security and safety. Therefore in order to investigate which concepts are likely to be benefitted from each other or make interplay with we compared security and safety domain models by identifying their similarities and differences. Concepts we think that have similar goals from security and safety domains are compared. The table below (See Table 3) gives an overview of similarities and differences between these concepts.

In Table 3, similarities and differences between security and safety domain models compared. Considering the Table 3, we can say that although some concepts have different names in security and safety domains they have similar goals and can be compared such as (i.e., Event), (i.e., Harm). And there are some concepts, which exist for security and do not exist in SDM such as (i.e., Threat agent), (i.e., Attack method). Also some concepts have no difference in comparison such as (i.e., Asset). Regarding this comparison we could say that in terms of asset model security and safety domain models can be used in common because they both are concerned with to protect what is valuable from harm for safety and impact for security. Although, there is a slight different between harm and impact as harm is actual damage and impact is potential, they are both connected to asset and can be modeled together.

Table 3. Similarities and Differences

Concept from Security	Concept from Safety	Similarities	Differences
Risk	Risk	-Consequences of both risks directly connected with asset. -Both of them are predictable up to some point.	-Security risk occurs due to an intentional negative incident on information system however safety risk occurs with accidental negative incidents.
Event	Accident	-Both of them are triggered with an incidents or incidents having negative impact on information system	-Event is combination of a threat and vulnerabilities however accident is combination of hazard and vulnerabilities.
Impact	Harm	-Both of them are occurring due to a specific solid reason. -Both causes value loss to asset.	-Impact is a potential harm to an information system. However, harm is specific actual damage.
Threat	Hazard	-Both of them are concerning potential future unplanned value loss occasions. -Both of them may cause value loss situations to the asset.	-Threat has agents having strategies to harm the asset (nothing happens randomly). -Threat has one state that is potential -However, hazard has potential and actual states. -And harms the asset with unplanned accidental occasions.
Vulnerability	Vulnerability	-Both of them are weaknesses increasing susceptibility of the system.	-No difference found.
Security criterion	Safety goal	-Both of them are setting the minimum required standards information system is looking for in order to protect the asset.	-Security criterion has security objectives safety goal has sub factors.
Asset	Asset	-Both of them are valuable -Both of them need protection for unplanned value loss occasions. -Both of them have the same sphere of stakeholder influence.	-No difference found.
Risk treatment	Safety policy	-Both of them are concerning about managing the risk/s. -Actions taken for both of them are based on written standards and have boundaries.	-A software engineer can specifically create definition and appliance of risk treatment for an information system. However, safety policy involves political, economic and moral decisions outside the decision-making realm of the software engineer.
Security requirement	Safety requirement	-Both of them are quality requirements. -Both of them are specifying what needed in order to protect the asset.	-Security requirements have to meet qualities that are important to organization such as confidentiality, integrity and availability. However, safety requirements have to meet specific safety policy.
Control	Safety mechanism	-Both of them have the intention to decrease chances of unplanned value loss. -Both of them have criterion on how to reduce the chances of unplanned value loss.	-No difference found.

2.4 Summary

In this chapter, we have presented security and safety domain models along with their concept definitions and samples to these concepts. For security we have presented ISSRM and for safety we have adopted a model from (Firesmith, 2003). We have also presented correspondence between security and safety domain models by focusing on their similarities and differences.

CHAPTER 3. Modeling Languages for Security and Safety Risk Management

This chapter introduces, modeling languages for security and safety risk management. Among the security risk management languages, Secure Tropos, MAD and Misuse Case together with its alignment to ISSRM introduced. Among safety risk management languages, HAZOP, PHA, FMEA and FTA along with its methodology and introduced.

3.1 Security Risk Oriented Modeling Languages

Security risk oriented modeling languages helps us for elicitation and validation of security requirements, they expose us to view hidden requirements. Some of these languages are based on textual descriptions, some of them are with diagrams and some of them covers both textual and diagram representations. In our study we introduced most frequently used security risk oriented modeling languages such as Secure Tropos, MAD and Misuse Case (Raspotnig, et al., 2013).

3.1.1 Secure Tropos
Description: Tropos is a software development methodology, which uses agents' concept through four development processes that are early and late requirements, architectural and detailed design (Bresciani et al., 2004). Secure Tropos is security-based extension of tropos that allows modeling security sensitive scenarios from the early stages of the development process (Matulevičius et al., 2012).

Structure: Secure tropos supports analysis of security through development processes based on different models. Security enhanced actor model analysis actors of the environment, actors of the system and dependency relationships between these (Matulevičius et al., 2012). Secure Tropos has actors, goals, soft goals, hard goals, resource and plan. Considering security related concept it has security constraint and thread. There are also dependencies between elements with dependee and depender. In it is detail an actor has intention and goal regarding the system. Their role represents the characteristics of an actor. These actors have soft goals and hard goals and in general the basic difference between these two can be described as; soft goal captures non-functional requirements and hard goal captures functionality requirement of a system (Matulevičius et al., 2012). In Tropos different alternative tasks to achieve a specific goal are modeled. There might be dependencies between users such as to achieve one goal a user may need another user to execute a task.

3.1.2 Mal-Activity
Description: MAD are the usage of unified modeling language (UML) diagrams with the exact same syntax and semantics to cover security aspects of systems in early development phases with some additions to UML (Sindre, 2007).

Structure: MAD have some additions to UML diagrams to capture negative scenarios as following: malicious activities are shown with icons that are inverse of normal activities, malicious actors are shown using white text on black background and their names are written as inverse where they have indicated swim lanes, malicious decision boxes shown as inverse of normal decision boxes and they are liable with making the best decision for malicious purpose.

3.1.3 Misuse Cases

Description: Uses cases are one of the most frequently used elements during requirements engineering, however they offer very limited for elicitation of security requirements (Sindre et al., 2004). That is why we there exist an approach called misuse cases to extend traditional use cases to also cover security requirements. In order to work with misuse cases there are five steps listed as follows; identify critical assets, define security goals, identify threats, identify and analyze risks and define security requirements respectively. In our study we are going to use misuse cases among security risk identification techniques because of the fact that it has detailed and well structured textual representation and frequently used technique in empirical studies (Raspotnig, et al., 2013) and most important of all has alignment with ISSRM (Soomro, 2012).

Structure: Use cases have two basic elements such as use case and actor. Similarly misuse cases have two basic elements as well such as misuse cases and misusers. Misuse cases can be described as consequences of series of actions with some certain harm done by misusers and misusers can be described as actors intentionally or accidentally making misuse cases scenario real. Diagram use cases relationships are include, extend, mitigate, threaten or exploits. Diagram shows how the misuser can cause harm to the system using the following elements such as actors, cases and their relations with each other. There also exist textual representation for misuse cases.

3.2 Alignment of Misuse Cases to ISSRM

The main reason for this alignment is to see the capabilities of misuse cases with the correlation of ISSRM. Following with this we may have a chance to identify security concerns at an early stage of software development and see potential security problems from a different point of view (Soomro, 2012). This alignment covers core concepts of ISSRM such as asset-related concepts, risk-related concepts, risk-treatment related concepts and risk management process. In our study, we apply our scenario of car theft *(Section 2.1.2)* to show how misuse cases can be for security risk management illustrating both textual and graphical representations with a running as follows.

3.2.1 Textual Representation Sample: Various templates have been suggested for the textual description of misuse cases but in our running example we will use the template suggested by (Sindre, et., 2001). Graphical representation gives the overall view of the requirements. However, textual representation captures the real essence of use cases. They also encourage developers to explain the story with plain sentences. There exist misuse case templates as lightweight and extensive (detailed) (Sindre et al., 2001). In our study we used the detailed textual representation deviated from (Kulak, et al., 2000) by (Sindre et al., 2001). See Table 4 for the example.

3.2.2 Graphical representation sample: Illustrates the interaction between *actor* and *misuser* with the system and specifications of set of actions performed. Fig 3, Fig 4, Fig 5 uses graphical representation of misuse cases. Diagrams uses all legends of use cases except *extends* and some special legends as *misuse case, threatens, mitigates, exploits security criteria* and *misuser*.

Asset model

In Fig 4, we present the context of car theft modeled in a use case diagram together with security criterions. According to ISSRM domain model we considered "drive car" as business asset,

"ignition start" as IS asset and confidentiality of the car as security criterion. The example focuses on "driver" and "car thief" who are getting into interaction with the asset that is the car. *Business use case* (i.e., drive the car) *includes* (i.e., Start the engine). S*ecurity criterions* represented in hexagons (i.e., Confidentiality of the car, Integrity of the car) are *constraints of, business use case* (i.e., drive the car).

Table 4. Example of the detailed misuse case template

Name	Steal the car.
Summary	Stolen car from that is company property.
Author	Servet Kurt
Date	14.03.14
Basic path	**bp1**: Car thief breaks the window of the car. **bp2**: Opens the door. **bp3**: Gets in to the car. **bp4**: Slots a screwdriver into the ignition and turn over it like a regular key. **bp5**: Starts the engine. **bp6**: Runs away with the car.
Alternative paths	**ap1:** Doors of the car unlocked. No breaking the window of the car is necessary. (Changes step bp1). **ap2**: Window of the car is open. No breaking the window of the car is necessary. (Changes step bp1).
Capture points	**cp1:** Car alarm starts ringing when the window of the car is broken (in step bp1) **cp2**: Engine does not start because there is immobilizer (in step bp2)
Mitigation points	**as1**: Toaster can not be moved away from kitchen worktable because it is tied up to worktable (in bp5), (extension point ext1)
Extension points	**[..]**
Trigger	**tr1**: Always true, this can happen anytime.
Pre conditions	**pr1**: Car does not have car alarm. **pr2**: Car does not have immobilizer.
Assumptions	**as1**: Car thief's screwdriver fits with the ignition (for all paths).
Worst case threat	**wc**.1:Car thief stole the car and run away.
Mitigation guarantee	**mg1**.:Immobilizer avoids car thief to start the engine with hot wire technique. **mg2**: Car alarm rings when the window of the car is broken.
Related business rules:	**br1**.: Company cars belongs to company and only authorized personnel can use them. **br2**: According to company rules during the work hours specified personnel has all the responsibility of the car.
Potential misuser profile	Experienced car thief who has knowledge on what type of cars may have immobilizer and hot wire technique.
Stakeholder and threats:	**Company**: Loses money if the car has stolen and maintenance works company does delays. **Driver**: Company's trust decreases to the employee. May lose the job.
Scope	Entire business
Abstraction level	Goal of the car thief.
Precision level	Focused

3.2.2 Graphical representation sample: Illustrates the interaction between *actor* and *misuser* with the system and specifications of set of actions performed. Fig 3, Fig 4, Fig 5 uses graphical representation of misuse cases. Diagrams uses all legends of use cases except *extends* and some special legends as *misuse case, threatens, mitigates, exploits security criteria* and *misuser*.

Asset model

In Fig 4, we present the context of car theft modeled in a use case diagram together with security criterions. According to ISSRM domain model we considered "drive car" as business asset, "ignition start" as IS asset and confidentiality of the car as security criterion. The example focuses on driver and car thief who are getting into interaction with the asset that is the car. *Business use case* (i.e., drive the car) *includes* (i.e., Start the engine). S*ecurity criterions* represented in hexagons (i.e., Confidentiality of the car, Integrity of the car) are *constraints of, business use case* (i.e., drive the car).
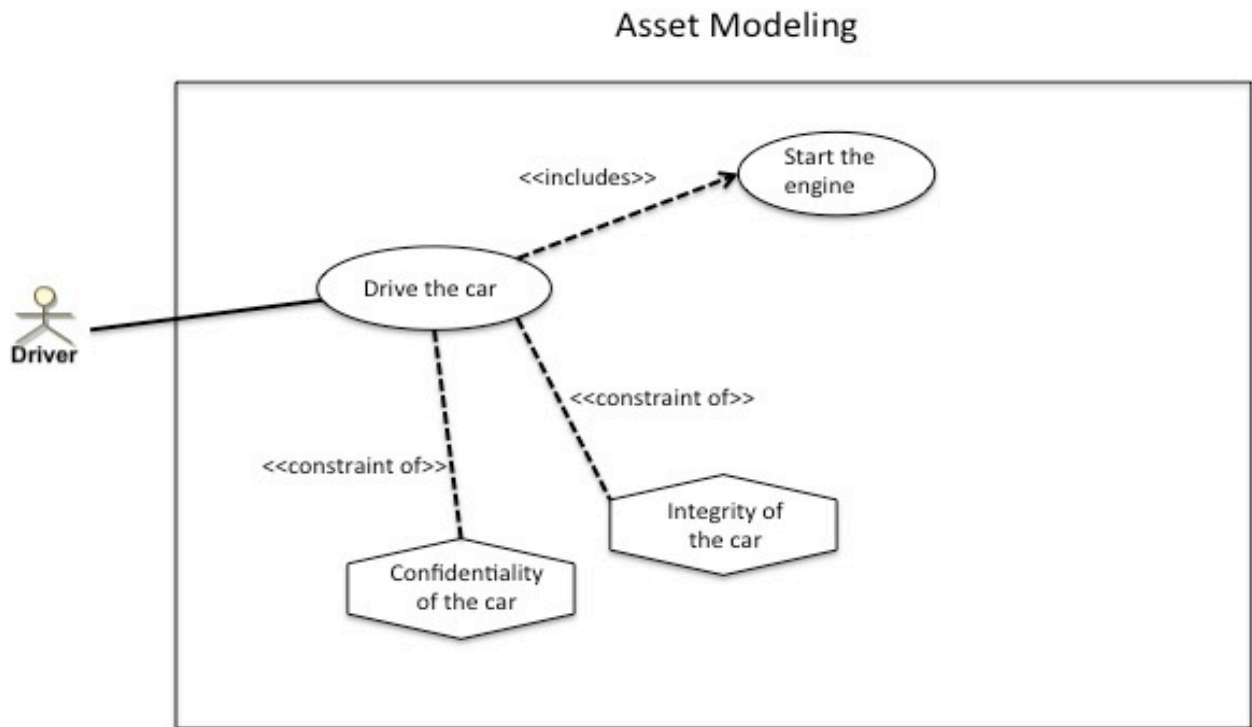
Asset Modeling

Fig 3. Asset Modeling

Risk model

In Fig 4, we present potential threat scenario. *Misuser* (i.e., Car Thief) initiates a misuse case (i.e., *Hot wire the ignition* includes *run away with the car*) by *exploiting* the *vulnerability* (i.e., Unsecure ignition start) that *includes, IS asset* (i.e., Start the engine). In Fig 4, the vulnerability is represented by filled grey use case. *Misuse case* (i.e., Hot wire the ignition) *threatens* the *IS asset* (i.e., Start the engine) and *leads to* (i.e., Car thief run away with the car) that *negates* the security criterion (i.e., Confidentiality of the car). The *threat* (i.e., hot wire the ignition) *leads* an *impact* on *business asset* (i.e., Drive the car).
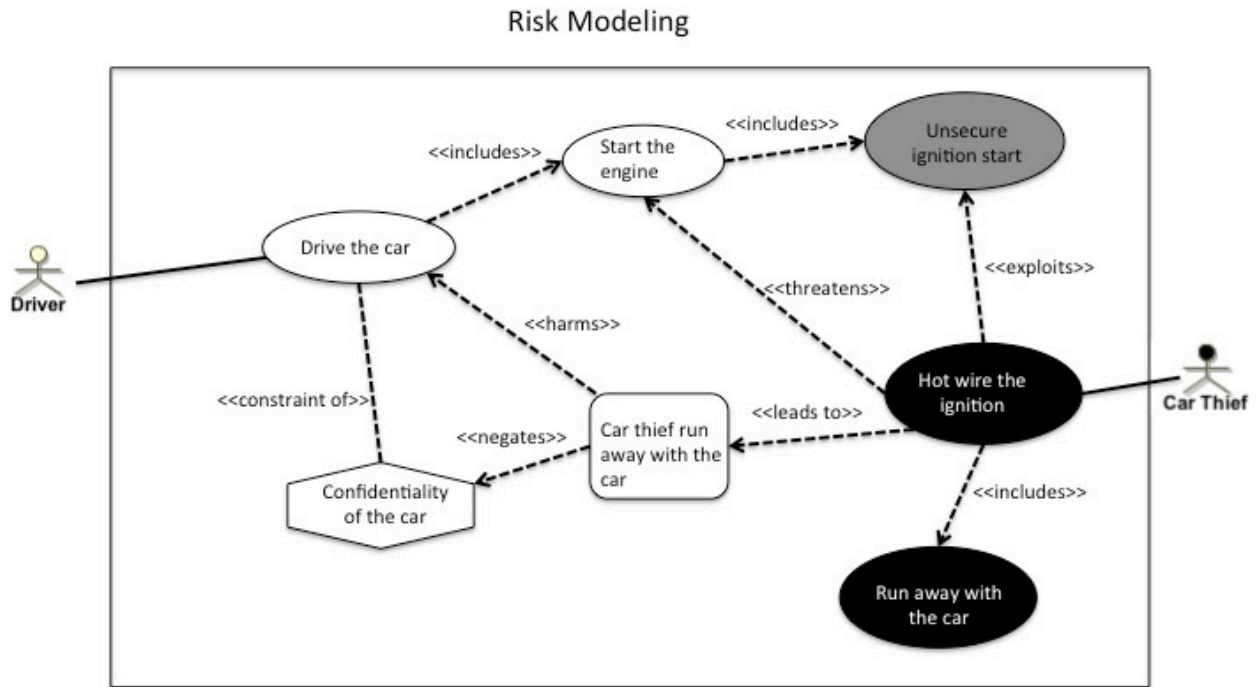
Fig 4. Risk Modeling

Risk treatment model

ISSRM domain model supports the risk treatment, control and its implementation. However, security risk oriented misuse cases do not support the modeling of these concepts. So security requirement is modeled as a *security use case* (Soomro, 2012). Security use case is represented with a lock inside (see Fig. 5). In Fig. 5, we present the security requirement for identified threats in our example. The *security use case mitigates* the *misuse case* (i.e., Hot wire the ignition). It ensures *security criterion* (i.e., Confidentiality of the car) *imposed* by *business use case* (i.e., Drive car)**.**
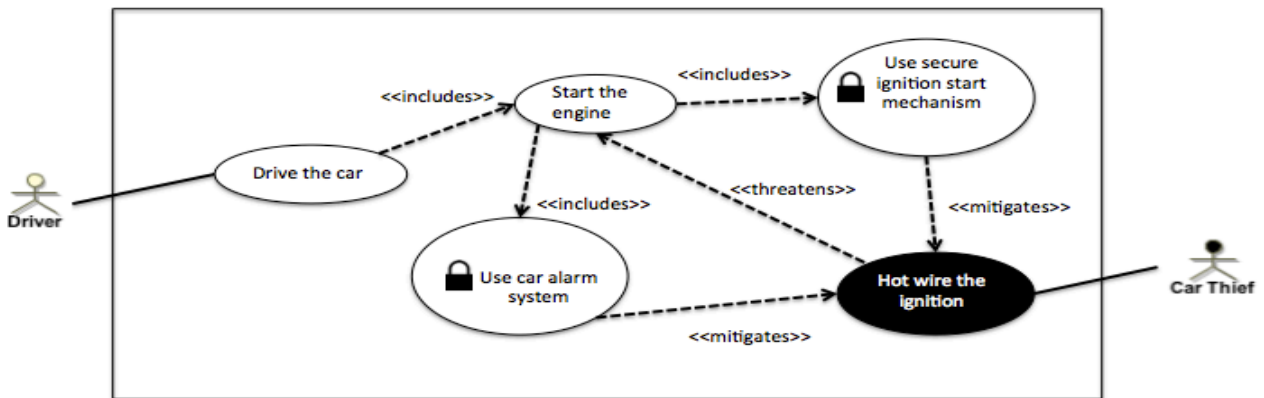

Fig 5. Security Requirement Modeling

3.3 Safety Risk Management Languages

Dependence on programmable equipment is increasing and these equipment the ones such as transportation and control system are under risk of physical injury or environmental damage that are aggregated as safety related risks (Sindre, et al., 2004). In order to identify these risks there exists some safety management languages. In our study we introduced the most frequently used safety risk identification techniques, which are HAZOP, PHA, FMEA and FTA (Raspotnig et al., 2013).

3.3.1 Hazard and Operability

Description: HAZOP is a method, which identifies hazards and problems to provide efficient operation. It indicates both hazards and operational threads in a system and makes analysis of them. At the same time HAZOP is a technique, which provides an approach to people letting them use their imagination and do iterative work by giving opportunity to team members stimulate each others ideas and build upon each other (Kletz, 1999).   Also by this way chance of missing something is being reduced.

Structure: Analysis are being held using guidewords in a textual way such as *none, more of, less of, part of, more than (as well as), other than*. There exist deviations generated by each guideword as it is shown in the *table 4* below. With these guidewords and their deviations generated analyzers of a team asks iterative questions. Same procedure applies with same questions to the next guideword. In order to avoid missing a detail analyses perform in a systematic way, so each sort of hazard considered in turn (Kletz, 1999).

Table 5. Deviations Generated by Each Guideword are taken from (Kletz, 1999)

Guide word	Deviations
None	No forward flow when there should be
More of	More of any physical property than there should be
Less of	Less of any physical property than there should be
Part of	Composition of system different from what it should be
More than	More components present in the system than there should be
Other than	What else can happen apart from normal operation

3.3.2 Preliminary Hazard Analysis

Description: PHA is a tool to elicit initial safety requirements in system, in other words potential or suspected hazards that may affect the design safety when the actual detailed design is not available yet. PHA identifies safety issues such as hazards, their casual associated factors, effects, level of risk and methods to mitigate these and affects the design safety (Ericson, 2005).

Structure: Starting point for PHA is analyzing hazards in detail concurrent from preliminary hazard list (PHL), which is collection of identified hazards. For example if a system is using gas generators analyst checks the PHL and obviously finds gas generators as hazardous element and

that gas generators explosion is a potential mishap together with many hazards. Next step is checking hazard checklist, which is a collection of known hazardous items and undesired mishap checklists. Afterwards, collaboration of these lists as PHL, hazard checklist and mishap checklists with system design PHA worksheets are created. These worksheets have columns such as hazard, causes, effects, recommended actions and comments. When these worksheets are done PHA report is created for identified risks, mishaps, hazards and mitigation methods (Ericson, 2005).

3.3.3 Failure Mode and Effect Analysis

Description: FMEA is a reliability procedure that analysis every possible failure in system design based on its ground rules. There exist four types of FMEA s as System FMEA, Design FMEA, Process FMEA and Service FMEA (Stamatis, 1995).

1. System FMEA analyzes systems and subsystems in early design stages.
2. Design FMEA analyzes products before they start manufacturing.
3. Process FMEA analyze manufacturing and assembling processes.
4. Service FMEA analyzes services before they received by customer.

Structure: FMEA has step-by-step bottom up approach. Meaning that failures of all components are being identified during FMEA procedure and their effects on overall system, which is creating, undesired events. Failure mode stands for component, service, procedure and design fail. They can be any errors or defects resulting in harm to the system, design, service or procedure. In FMEA failures can be prioritized according to their effects. Effect analysis stands for consequences of failures. Local effect stands for failure effect that is happening to the item under analysis. System effect stands for overall effect that is consequence of local effect. Corrective action stands for prevention ways to failure modes. FMEA is used for identifying actual risks on running system for continuous improvement or risks of failures to lower the chances of failures or prevent (Nancy, 2004).

3.3.3.1 FMEA Running Example

Sample scenario is taken from (Nancy, 2004), which is about a bank's ATM system. Table 6, illustrates the use case "Dispense Cash" and failure modes related with that function.

In order to create this table for the given function "dispense cash" following steps performed are performed:

1. Understand the scope of system and purpose of the function wants to be analyzed which is "Dispense Cash" in our scenario.
2. Identify type of FMEA. For our scenario FMEA identified as "service" for the function "dispense cash" among four different types of FMEA.
3. Create FMEA table (See Table 6)
4. Identify in what ways "Dispense Cash" may fail. These are about potentially in what ways a failure could happen. We call them as failure modes.
5. Identify what are the local and system effect of the failures. Meaning that what kind of consequences will come up with these failures, locally and consequences of local failure on system.
6. Identify corrective actions to prevent failure modes.

Table 6. FMEA Example for "Dispense Cash" adopted from (Nancy, 2004).

Unit	"Dispense Cash"		
Failure mode	**Local effect**	**System effect**	**Corrective action**
Does not dispense cash	Customer dissatisfied.	Bank loses customer.	• Internal low cash alert. • Loading procedure (riffle end of stack) • Increase bandwidth to decrease heavy network traffic.
Dispenses too much cash	Bank loses money.	Bank goes bankruptcy.	
Takes too long to dispense cash	Customer gets annoyed.	Bank loses customer.	

Table 7 FMEA and SDM

SDM		FTA and its Alignment to SDM		FMEA
		FTA element	**Misuse case diagram**	**FMEA element**
Asset	*Asset*	-	Use case	-
	Safety goal	-	Safety element	-
	Vulnerability	-	-	-
Risk	*Safety risk*	Fault Tree		-
	Accident	Basic event	-	System Effect
	Hazard	Intermediate event, undeveloped event.	-	Failure Mode
	Vulnerability	-	Vulnerability use case	-
	Harm	Intermediate event.	Impact element	Local Effect
Risk treatment	*Safety policy*	-	-	-
	Safety requirement	-	Safety use case	Corrective action
	Safety mechanism	-	-	-

3.3.3.2 FMEA and SDM

There exist no alignment of FMEA with SDM. However, we have proposed alignment of FTA with SDM and considering the fact that FTA supplements FMEA (Stamatis, 1995) and their relationship in terms of the literature they rely on are common we have created a table (See Table 7) showing FTA elements with SDM and engagement of FMEA to it.

Table 7 shows that FTA and FMEA both elicits accident, hazard and harm. A part from this FMEA covers safety requirements that in our alignment we have covered it using Misuse Case. But safety risk that FTA covers is not covered by FMEA. Regarding our validation for FTA and its alignment to SDM, as also Table 7 shows that it is possible to make comparison for its correctness because Misuse Case and FTA is already aligned with SDM and a risk analysis made by FMEA and Misuse Cases can be compared with FTA and its alignment to SDM since FTA supplements FMEA and literature they rely on are common (Stamatis, 1995).

3.3.4 Fault Tree Analysis

Description: FTA is a top down, deductive failure analysis using Boolean logic to expose undesirable states of system and combine it with series of lower-level events. FTA uses to understand in what ways a system can fail, what are the risks and what are the best ways to reduce these risks. General implementation area of FTA is aerospace, chemical and nuclear power industries (Brooke et al., 2003).

Structure: FTA constructs a tree starting with initial undesired event at the root and connections of either primary or intermediate events with gates to each other. There exist three main construct elements as primary fault symbols, gate symbols and transfer symbols and their sub elements (See Fig 6.).
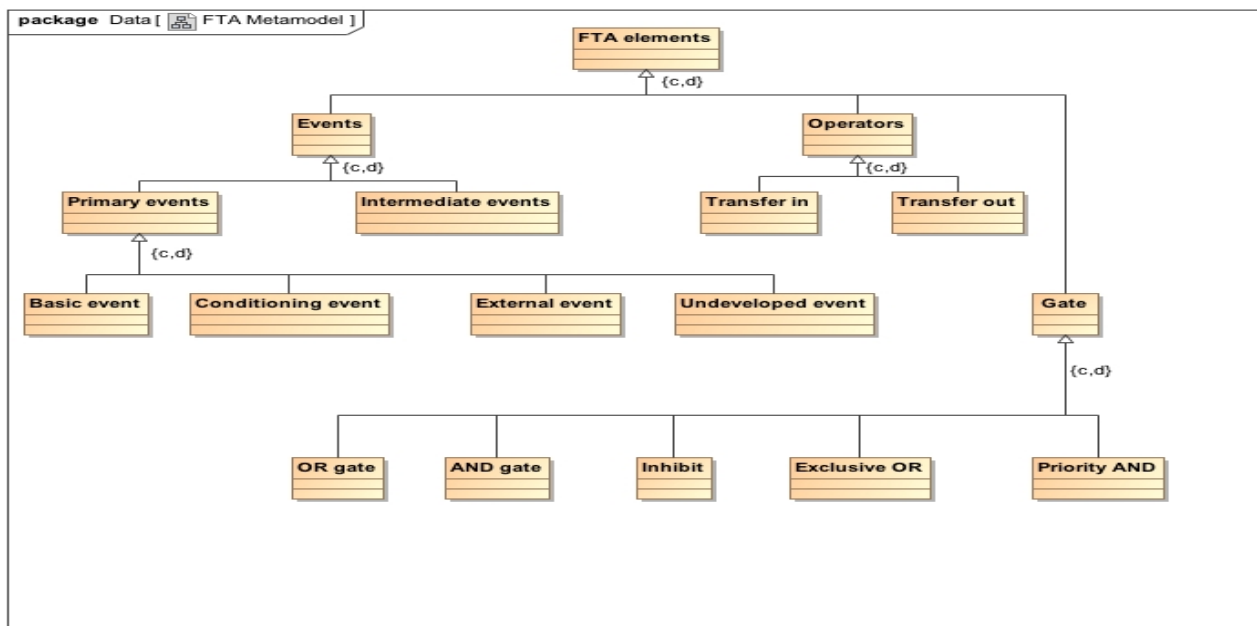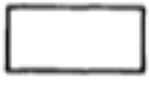

Fig 6. FTA Metamodel

Some of the key elements of FTA are *and, or, conditioning event, undeveloped event* and *intermediate event*. AND describes logical operation where all input events are necessary to have

the output event. OR describes logical operation where only one input event is necessary to have the output event. CONDITIONING EVENT is basic fault input, which requires no further development and branching (end branches of the fault tree). UNDEVELOPED EVENT is a basic fault event that is not developed its causes. INTERMEDIATE EVENT is fault event that is caused by failures and needs further developments (branching).

Table 8. Fault Tree Symbols

◯	Basic event	⬭	Conditioning event	◇	Undeveloped event
⬠	External event	▭	Intermediate event	⌂	AND
⌂	OR	⌂	Exclusive OR	⌂	Exclusive AND
⬡	INHIBIT	△	Transfer IN	△	Transfer OUT

3.3.4.1 General Steps For Fault Tree analysis

A Fault tree should only be created when the analysis of the system is done and it is understood how the system functions.

Steps

1- First analyst defines the system and it's boundaries. It explains initial condition of system that is needed for failure information.

2- Define the undesired event to be analyzed. This should be a problem of interest that analysis will address such as a particular system failure.

3- Determine the treetop structure such as with intermediate events. These will be the events that directly lead to the top event.

4- Determine the events that directly lead to each intermediate event until the fault tree model is complete and explore the fault tree for the combinations of events contributing to initial undesired event. By determining the immediate, necessary and sufficient causes for the occurrence of the top event.

5- Perform quantitative analysis.

There needs to be made a probability calculation in this step. However, for computerized systems especially for distributed systems it is difficult to assign useful probabilities to the events due to

the fact that these systems have discrete, non-linear nature (Brooke et al., 2003). Instead analyst should carry out the risk analysis and observe how the system may fail.

6- Interpret results for identifying vulnerabilities in the system in order to reduce the risks that are associated with these risks.

3.3.4.2 Fault Tree Construction Ground Rules
When we determine about a specific problem to create fault tree the following below rules should be applied to explain what such events actually are and how they should be structured in the tree. These ground rules are suggested for fault tree construction by (Fault Tree Handbook, 1981).

Ground Rules

*1-*Write the statements, which will be entered in the event boxes as faults by stating precisely what the fault is, and when it occurs.

*2-*Examine these boxed statements and ask the following question: "Can this fault consist of a component fault?" If "yes" then classify this event as "state of component fault" and use Ground Rule 2.1 if "no" then classify this event as "state of system fault" and use Ground Rule 2.2

   2.1 Add an OR gate below the event and look for primary, secondary and command modes.

   2.2 Look for minimum immediate, necessary and sufficient cause or causes. That may branch with an AND, OR, INHIBIT or no gate at all.

*3-*No gate-to-gate relationships are possible. Analyst has to put an event between two gates.

*4-*No miracles rule. The normal functioning of a component propagates a fault sequence, then it is assumed that component functions normally. Meaning those things that would normally occur as the result of a fault will occur, and only those things.

*5-*Complete the gate rule. All inputs to a particular gate should be completely defined before further analysis of any of them undertaken.
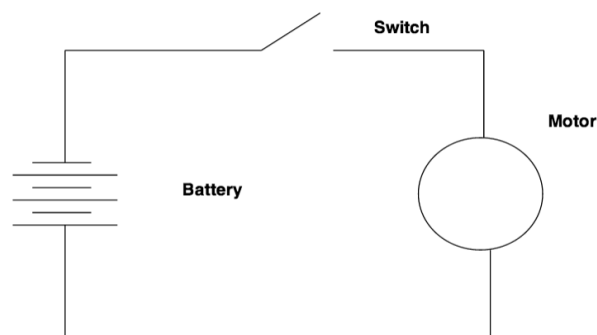
3.4 Fault Tree Analysis and Construction Sample

Fig 7. Simple Battery Powered Circuit taken from NASA HQ Bill Vesley

Above steps given for FTA and construction rules are followed in the running example of simple battery powered circuit (See Fig. 7).

Steps

1- Intended Function: Motor is used for some purpose that is not known.

Physical Boundaries: Battery.

Analytic Boundaries: Switch, Motor, Battery, transition elements.

Initial Condition: Switch open. Motor does not work.

2- Top event: Motor does not start when the switch is open.

3.5 Summary

In this chapter, we have presented different modeling languages for security and safety risk management along with their descriptions and plain structures. For security Secure Troops, MAD, and Misuse Case and its alignment to ISSRM presented. For safety HAZOP, PHA, FMEA and FTA presented. For FTA, we have also presented its meta-model (See Fig 6.), diagram elements (See Table 8), methodology and a construction sample. Methodology we have presented for FTA includes its ground rules and general steps needs to be followed in order to apply it. We also presented application of FTA with a running example from NASA HQ Bill Vesley "Simple Battery Powered Circuit". Next chapter presents our main contribution, which is alignment of FTA with SDM.
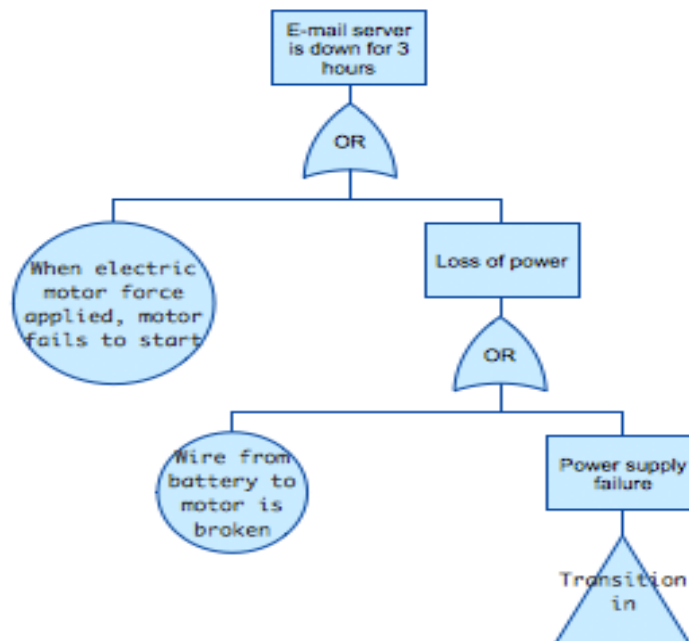

Fig 8. Battery Powered Circuit Fault Analysis adopted from NASA HQ, Bill Vesely

CHAPTER 4. Alignment of Fault Tree Analysis to SDM

The main reason for the alignment of FTA to SDM is to see its capabilities with SDM. Following with this alignment we may have a chance to identify safety concerns at an early stage of software development and see potential safety problems from a different point of view. This alignment covers core concepts of SDM such as asset-related concepts, risk-related concepts, and risk-treatment related concepts.

4.1 FTA Running Example

This study applies FTA and its alignment with SDM for a company using their own mail server (Marquis, 2008) by following the steps defined separately for asset model, risk model and risk treatment model (Section 4.2.2, 4.2.3, 4.2.4). As in the methodology also example split to three parts as asset model, risk model and risk treatment model.

4.1.1 Asset Model

In Fig.9, we present the context of an employee trying to send an email to a customer using company's mail server in the modeled use case diagram. Asset model is the use case of the *intended function* (i.e., Send an E-mail to Customer) wants to be performed by *actor* (i.e., Employee) and it *includes* (i.e., Use Mail Server) (see Fig.9). *Safety goal targets safety level* (i.e., System adaptation), since there exist no diagram element for safety goal we illustrated it with octagon. The employee and the mail server are characterizing the *asset* (see Fig.2) of the system by employee being external to the accidental system and mail server as being part of the accidental system. Since employee is external to the accidental system we analyze boundaries of mail server that is being part of accidental system as hardware and software.

Fig 9. Asset Model
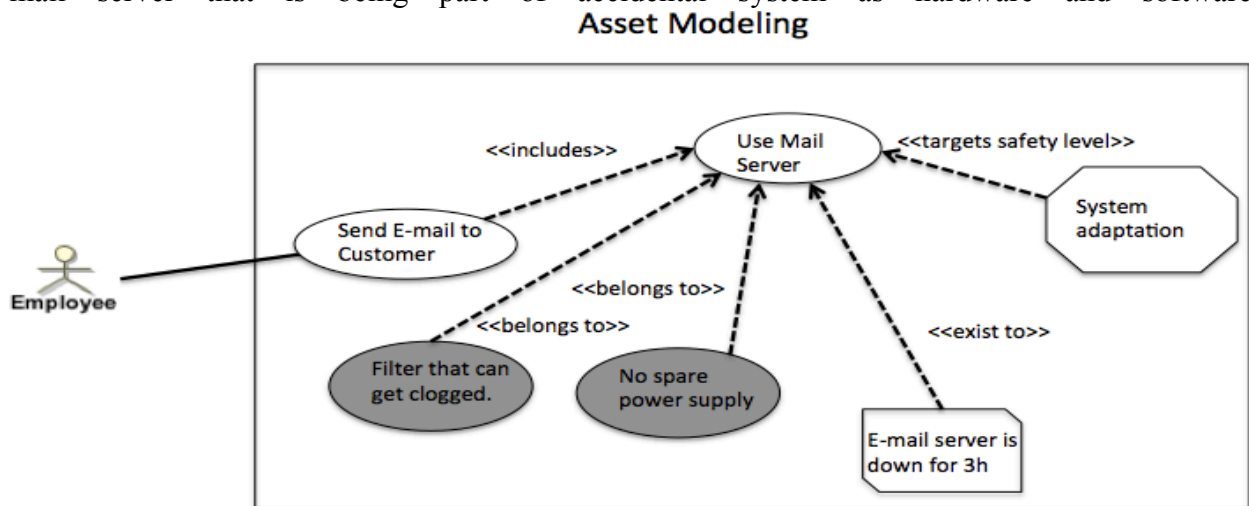
In our example system boundary (hardware) has vulnerabilities that *belongs to* asset to such as (i.e., no spare power supply) (i.e., filters that can get clog) that causes to system failure when it exploits and the *harm (i.e.,* mail server down for 3 hours in this example) causes is the system failure to be analyzed with risk modeling using FTA (see Table 9), since there exist no diagram

element for harm we created a new diagram element for it and illustrated with snip diagonal corner rectangle.

Table 9. Failure & Vulnerability Analysis

Failure Analysis	Failure Mode	Vulnerability
Employee unable to send e-mail	Power supply failure	-No spare power supply. -Filter that can get clog.

4.1.2 Risk Model

In Fig. 10, fault tree presents potential accidental scenario to the asset modeled for the use case in Fig 9 regarding the intended function (i.e., Send an email to a customer). Initial undesired event at the top of the tree is *harm* (i.e., Email server is down for 3 hours) asset gets. Intermediate event that is directly related to top event (i.e., Loss of power) is *hazard* to the asset also undeveloped event (i.e., Software failure) is *hazard* that is also directly related to top event. The branches of the tree, intermediate event (i.e., Loss of power) branches with one undeveloped event that is a *hazard* (i.e., No spare power supply) and one intermediate event that is harm (i.e., Power supply failure). And finally FTA completes with actual hazard that is an *accident* (i.e., Filter clogged).
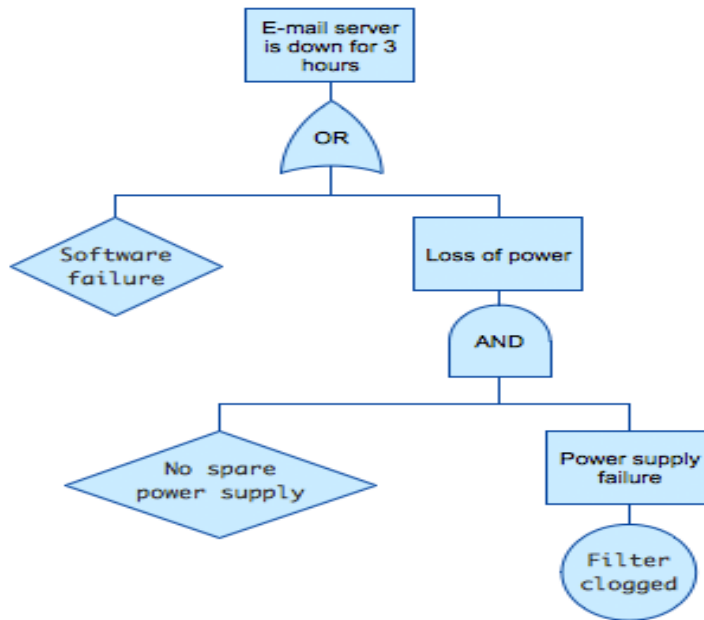
Fig 10. Risk Model

4.1.3 Risk Treatment Model

SDM supports safety policy, safety mechanism and its implementation. However, FTA and its alignment to SDM do not support the modeling of these concepts but safety requirement as a safety use case. Safety use case is represented as an attention symbol (see Fig 11). In Fig 10, we present fault tree of present accidental scenario to the asset. The use case (i.e., Use Mail Server) *includes* safety requirement (i.e., Maintain filter unclogged). Safety requirement *eliminates* the *harm* (i.e., E-mail server is down for 3h). And it ensures that email server will not be down accidentally due to clogged filters and our *elimination* of safety risk will *establish* adaptation ability to our system that is *safety goal*.

Safety Requirement Modeling

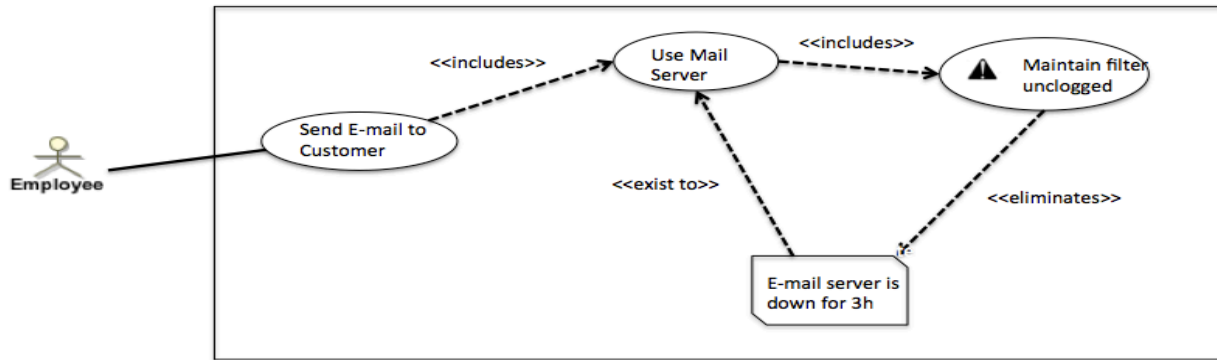**Fig 11. Safety Requirement Model**

4.2 Concept alignment of Misuse Cases and FTA with SDM

To align FTA to SDM we created steps to be followed for each of these asset-related concepts, risk-related concepts, and risk-treatment related concepts. In some concepts we used helping hand of use cases technique. Also we neglected probability analysis of FTA from our alignment since for computerized systems especially for distributed systems it is difficult to assign useful probabilities to the events due to the fact that these systems have discrete, non-linear nature (Brooke et al., 2003). Instead we focused on how to carry out risk analysis, asset modeling, and risk treatment and how a system may fail.

4.2.1 Misuse Case and FTA within the concept of SDM

There exist no alignment of misuse cases or FTA with SDM. In this section we also describe the alignment of misuse cases and FTA with the concepts found in SDM. In Table 3 we made a comparison between similarities and differences between ISSRM and SDM and due to similarities illustrated in Table 3 we think that concepts from ISSRM that are already aligned with misuse cases can be used for similar concepts from SDM. Table 10 shows the alignment of misuse cases for security domain model made by (Soomro, 2012) to ISSRM and matches of concepts with SDM also how it aligned for safety with cross pollinations between FTA and misuse cases to cover safety concepts. First column shows SDM similar concepts together, second column shows ISSRM concepts, third column shows actual misuse case diagrams used to represent these concepts, fourth column shows misuse case elements to represent matching safety concepts with ISSRM, fifth column shows FTA elements to represent concepts for safety, sixth column shows SDM concepts, seventh column shows similar safety concepts together. When we consider ISSRM domain model and misuse case we can see that it is well aligned with security domain model, in a way that most of the concepts for ISSRM are covered by misuse cases. Regarding the fact that SDM and ISSRM have similarities, and our idea to make cross-pollinations between techniques for safety and security we are going to use some misuse case diagram elements for SDM as it has shown in fourth column of Table 10. And for the rest of the safety concepts FTA will cover as it is showed in fifth column. To sum up, Table 10 will guide us which element to use when we switch to misuse cases for SDM, and which elements to use for FTA. And as the table shows by this way we will align the techniques we are planning to use for safety risk identification to SDM.

Table 10. Concept Analysis of Misuse Cases for ISSRM and SDM (NA- Not Applicable, concept not found)

ISSRM			SDM			
Domain Model		**Misuse Case**	**Misuse Case**	**FTA**	**Domain Model**	
Asset	Asset	Actor, use case.	Use case	-	*Asset*	**Asset**
	Security criteria	Security criterion hexagon.	Safety element octagon	-	*Safety goal*	
	-	-	Vulnerability use case	-	*Vulnerability*	
Risk	Security risk	Misuser, misuse case, vulnerability use case, use case, impact element.	-	Fault Tree	*Safety risk*	**Risk**
	Event	Misuser, misuse case, asset Vulnerability use case.	-	Basic event	*Accident*	
	Threat	Misuser and misuse case	-	Intermediate event, undeveloped event.	*Hazard*	
	Vulnerability	Vulnerability use case	-	-	-	
	Impact	Impact element	Harm element	Intermediate event.	*Harm*	
	Threat agent	Misuser	NA	NA	*NA*	
	Attack method	Misuse case	NA	NA	*NA*	
Risk treatment	Risk treatment	-	-	-	*Safety policy*	**Risk treatment**
	Security requirement	Security use case	Safety use case	-	*Safety requirement*	
	Control	-	-	-	*Safety mechanism*	

In Table 11, we show relationship of FTA's diagram elements and synonyms with SDM concepts. First column shows SDM similar concepts together, second column shows SDM concepts, third column shows FTA synonyms and fourth column shows FTA elements. Synonyms to SDM concepts for FTA found and extracted from (Fault Tree Handbook, 1981). As we can see in the table risk elements are covered very well since FTA is all about identifying all possible risks, faults, failures and failure ways. However, as it is expected in FTA there is no risk treatment terminology found. Also although asset mentioned as system, component participant like risk treatment there exist no way to model it with FTA. To sum up, from table 10 we can interfere that FTA produces terminology and ways focused on risk and these synonyms could be matched as it has shown in Table 10 with SDM concepts.

<div align="center">

Table 11. FTA and SDM

</div>

SDM		FTA	
		Synonyms	**FTA element**
Asset	*Asset*	System, component, participant.	-
	Safety goal	-	-
	Vulnerability	Failure ways.	-
Risk	*Safety risk*	-	Fault Tree
	Accident	Failures.	Basic event
	Hazard	Faults, failure mode.	Intermediate event, undeveloped event.
	Harm	Failure effect	Intermediate event.
Risk treatment	*Safety policy*	-	-
	Safety requirement	-	-
	Safety mechanism	-	-

Regarding the fact that misuse cases are already aligned with ISSRM and similarities of it studied with SDM (Table 11), we illustrated a new table (Table 12) that covers all concept for safety domain with diagram elements from both FTA and misuse cases by making cross-pollination between techniques. In Table 12, first column represents SDM concepts, second column represents FTA element and third column represents misuse case diagram.

Table 12. SDM with FTA and Misuse Cases

SDM		FTA	Misuse Case
		FTA element	**Misuse case diagram**
Asset	*Asset*	-	Use case
	Safety goal	-	Safety element
	Vulnerability	-	-
Risk	*Safety risk*	Fault Tree	
	Accident	Basic event	-
	Hazard	Intermediate event, undeveloped event.	-
	Vulnerability	-	Vulnerability use case
	Harm	Intermediate event.	Impact element
Risk treatment	*Safety policy*	-	-
	Safety requirement	-	Safety use case
	Safety mechanism	-	-

Coming to a conclusion from analysis we made (Table 10, 11, 12) we think that similar concepts from SDM and ISSRM are as shown in Table 13 and can be used for similar purposes with same or modified misuse case diagrams. Syntax to misuse case diagrams is given in Table 14 Table 15 and Table 16.

Table 13. Similar Concepts between ISSRM and SDM

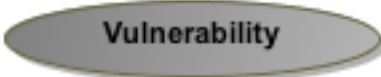| SDM	ISSRM	Misuse Case Diagram
Asset	Asset	Actor, use case
Vulnerability	Vulnerability	Vulnerability use case
Harm	Impact	Harm element
Safety requirement	Security requirement	Safety use case

4.2.2 Alignment of Asset Related Concepts
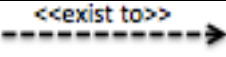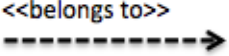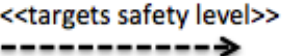
In Table 14, we introduced safety oriented misuse case syntax to represent the SDM asset related concepts. *Assets* in SDM correspond to Actor and Use Case in safety oriented misuse case that supports *includes* relationship. The *harm*, which is a failure effect and initial state of the system caused by accident, *exists to* asset, illustrated with visual syntax we modeled that is diagonal rounded corner rectangle and it supports *exist to* relationship. And *vulnerability* which *accident* exploits asset *has* modeled as a filled grey use case as it is in SROMUC syntax (Soomro, 2012) and it supports *has* relationship. *Safety goal* modeled asset has to carry modeled as octagon.

Steps needs to be followed for asset modeling are as below:

Steps

1- Draw the use case for intended function including *asset* that needs to be protected from *harm* in it.
2- Identify system boundaries; failure effect and vulnerabilities.
3- Show safety goal that targets safety level for asset for one of these safety sub factors; asset protection, safety incident detection, safety incident reaction or system adaptation.
4- Initiate harm exists to *asset*, which is a failure effect and initial state of the system, caused by *accident.*

Table 14. Asset Related Concepts (C- Concept, R- Relationships)

SDM	Type	Model Syntax with Misuse Cases
Asset	C	
Harm	**Belongs to risk related concepts but has a representation in asset model	
Vulnerability	C	
Safety goal	C	
Supports	R	
Exist to	R	
Belongs to	R	
Targets safety level	R	

4.2.3 Risk Model

In Table 15, we introduced FTA syntax to represent the SDM risk related concepts. *Harm* represented as failure effect and supports *causes* relationship, *hazard* represented as faults and failures and supports *due to* and *way result in* relationships, *accident* represented as failure mode and the resulting tree of these components as *safety risk*. These concepts supports relationships between each other using gate symbols and line connectors (See Table 8). Meta model for FTA is given in Fig 6. As it is shown in Table 15 FTA syntax column, syntax for representing accident is basic event, hazard is intermediate event, undeveloped event or basic event, failure mechanism is intermediate event or undeveloped event and harm is intermediate event that is also top tree element.

Steps needs to be followed for risk modeling and constructing the FTA is given below:

Steps

1- Place *harm* exist to asset that is undesired event to be analyzed to the top of tree as top event.
2- Determine intermediate events faults, failures (*hazards*) that are directly related to top event, which may result in *accident.*
3- Determine basic initiating faults, failures (*hazards*) that are directly related to each intermediate event until the FTA is complete. FTA completion means when *accident* defined and no further development is needed.

4.2.4 Risk Treatment Model

In Table 16, we introduced FTA syntax to represent SDM risk related concepts. *Accident* determined with basic event taken from FTA tree. *Safety risk* due to hazard way result in accident represented with use case that is a cause of harm. In order to illustrate *safety requirement* we created a safety use case by adding attention symbol in use case *and* it supports *eliminates or reduces relationship* to the *safety risk.*

Steps

1- Choose basic event, which is the *accident* from the FTA tree and think of safety requirement to eliminate the accident that causes harm.
2- Draw harm and show exist to relationship of it with asset.
3- Draw safety use case and add one of these four key words to it; maintain, achieve, cease or avoid representing *safety requirement* that has eliminates/reduces relationship with harm.

4.3 Summary

In this chapter, we have presented alignment of FTA with SDM. In order to identify fundamentals of this alignment, we have analyzed concepts of ISSRM and SDM for security and safety risk management languages we plan to use for our alignment, which are FTA and Misuse Case (See Table 10-11-12). We presented step-by-step methodology of our alignment along with their concept diagrams. We have also presented the alignment with a running example (See 4.1).

Table 15. Risk Related Concepts (C- Concepts, R - Relationships)
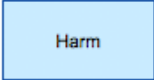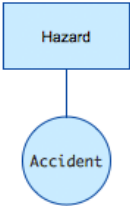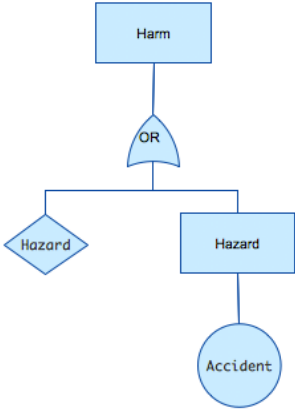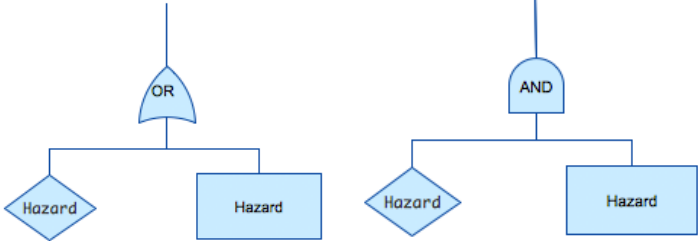
SDM	Type	Model Syntax for FTA
Harm	C	
Hazard	C	
Accident	C	
Safety Risk	C	
Vulnerability	C	-
Causes	R	-
Way result in	R	
Due to	R	
Exploits	R	-
Exists due to	R	-

Table 16. Risk Treatment Related Concepts (C- Concept, R - Relationships)

SDM	Type	Model Syntax with Misuse Case
Safety requirement	C	
Safety mechanism	C	-
Eliminates	R	
Reduces	R	
Fulfills	R	-
Specifies	R	-
Establishes	R	-
Elimination reduces	R	-

CHAPTER 5. CORRECTNESS OF FAULT TREE ANALYSIS ALIGNMENT WITH SAFETY DOMAIN MODEL

In this chapter, we validate our FTA alignment with SDM by examining the correctness of it. In order to examine correctness of our alignment, firstly we have gathered a scenario (Section 5.3) and extracted risk analysis models from this scenario then applied our alignment with it. Secondly, we presented a table that compares safety analysis done by Stålhane and us for boiler tank system (Stålhane & Sindre, pp. 425-426, 2007) and discuss the comparison. Finally we have lined up threads to validity.

5.1 Goal of the Case Study

The purpose of having this case study is to prove the correctness of our alignment for FTA with safety domain. This case study gives us chance for overview of our methodology so that we can check correctness of our method.

5.2 Validation Design

By assuming that the risk analysis done by Stålhane for the boiler tank scenarios is correct, we compare our FTA risk analysis with it and ask the validation question below:

Validation Question: Which safety risk analysis is better for the "boiler tank scenario" FTA and its alignment to SDM or FMEA?

In order to answer the validation question, we have created a validation design (Fig. 12) that follows three main steps. Firstly it extracts risk analysis done with FMEA and Misuse Case from the scenario (Section 5.3) and applies FTA. Secondly, we gather all the results together. And finally, we compare the results and create fundamentals for giving answer to validation question.
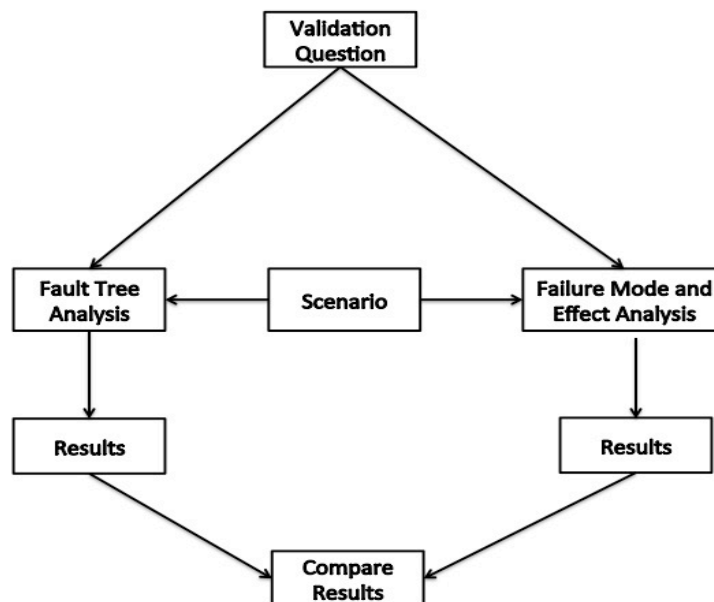

Fig 12. Validation Design

5.3. Case Study Scenario

We have needed a scenario that is applicable for our alignment that needs to be already applied for safety analysis with misuse cases or another technique that is tailored for safety. In our case FMEA and Misuse Case along with its diagram and textual representation. Therefore we picked the suitable scenario from (Stålhane & Sindre, pp. 425-427, 2007) as follows below:

"Human operator functions related to an automated system used to keep the water level in a tank constant while delivering steam to an industrial process. Filling the tank through one valve and emptying it through another valve when needed do this. If the pressure in the tank becomes too high, a relief valve should open automatically as the pressure exceeds the critical pressure pre-set by the operator. The operator may also manually empty the tank (for instance if the relief valve fails to work when the pressure becomes too high) or manually fill the tank (if the automatic adjustment of water level does not work)."

5.4 Model Extraction from Stålhane and Application of FTA with the Scenario

5.4.1 Misuse Cases

Fig 13. Shows a human operator whose aim is keeping the water level constant in a tank while delivering steam to the system by doing operations such as shown in use cases. In the Fig 13, it is also showed some misuse cases that could be done accidentally by the operator also system faults. These all misuse cases threatens on or more use cases in other words intended functions that wants to be accomplished by operator. The figure does not show threats or mitigations for a particular use case. Therefore there given textual representation of particular use case, which is "Empty tank manually". The use case "Empty tank manually" also analyzed with FMEA in the paper and will be analyzed by us with FTA for techniques to be compared.
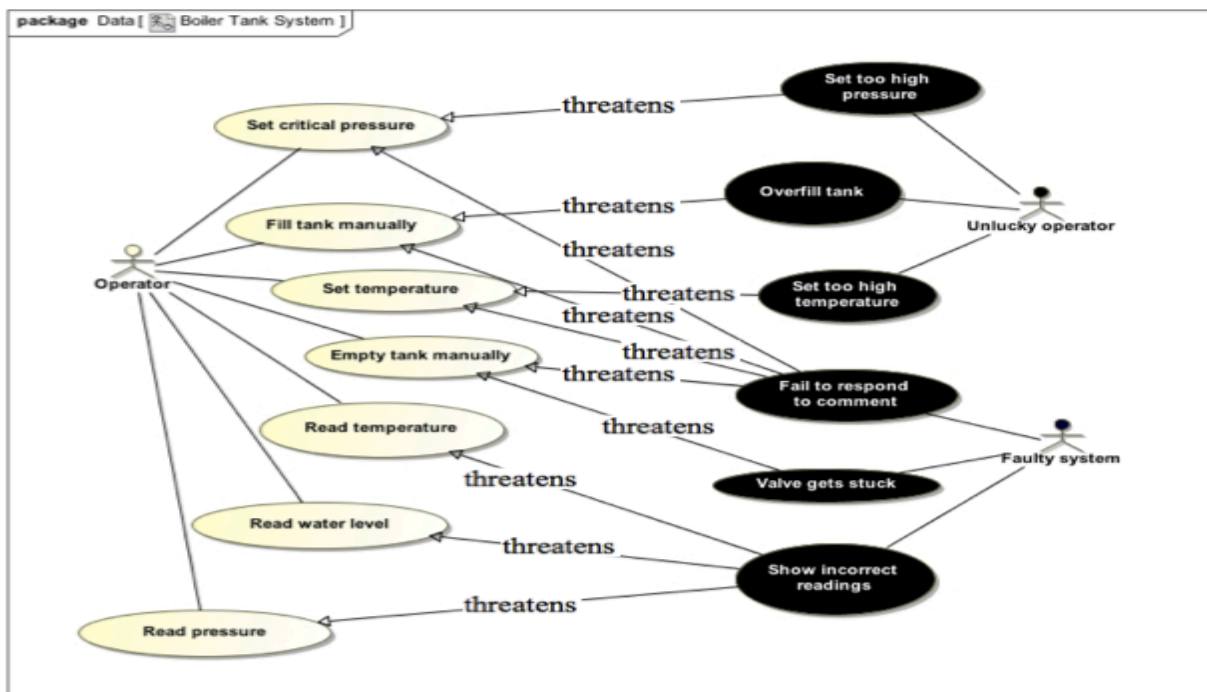

Fig 13. Sample Safety Oriented Misuse Case Diagram for a Boiler Tank System taken from (Stålhane & Sindre, pp. 426, 2007)

The textual representation shows (See Table 17) user actions, system responses, threats and mitigations for the use case empty tank manually and also exceptional paths that can be followed to achieve empty tank manually. Textual representation gives detailed risk analysis of particular use case and will be fundamental part for comparison with FTA.

Table 17. Textual Representation of "Empty tank manually" Use Case adopted from (Stålhane & Sindre, pp. 427, 2007)

Use case name	"Empty tank manually"		
User actions	**System response**	**Threats**	**Mitigations**
	System alarms operator of high pressure	System fails to raise alarm; Operator fails to notice alarm	2 independent alarms; Use both sound and blinking lights
Operator issues command to empty tank		Operator fails to react (e.g., incapacitated?). Operator gives wrong command, e.g., filling tank	Alarm backup operator; Auto sanity check, disallow filling at high pressure
	System opens valve to sewer	System fails to relay command to valve; Valve is stuck	
Operator reads pressure		Operator misreads and stops tank emptying too soon	Maintain alarm blinking until situation normal
	Pressure returns to normal	This is not achieved, see exceptions	
Operator stops tank emptying and logs the event. This ends the use case.			
Exceptional paths			
	Opening valve is insufficient to normalize pressure		
Operator issues command to reduce temperature		Operator gives wrong command, e.g., increase temperature	Automatic sanity check, disallow temp increase at high pressure
	Pressure returns to normal		
Operator logs the event. This ends the use case.			

5.4.2 Failure Mode and Effect Analysis

In Table 18, consequences of failure mode collected under two sub titles as local effect and system effect. Local effects are related with the component have been analyzed for and system effects are related with overall system effects that component is part of. Corrective action is possible mitigations to weaknesses in the system in order to handle with failure mode that is a system unit not working properly.

Table 18. FMEA table for "Empty Tank Manually" Use Case adopted from (Stålhane & Sindre, 2007)

Unit	"Empty tank manually"		
Failure mode	**Local effect**	**System effect**	**Corrective action**
Valve will not open	Cannot empty tank.	Accident – too high pressure	• Valve status indicator • Duplicate valve • Must be possible to turn valve without motor
Valve will not close	Cannot fill tank	No steam delivered	

5.4.3 Fault Tree Analysis for Boiler Tank System

This study applies modeling with FTA for boiler tank system (Section 7.1) and illustrates the usage of FTA. It covers asset model (see Fig. 14), risk model (see Fig. 15), and safety requirement model (see Fig. 16).

Asset Model

In Fig. 16, we present the context of an operator trying to empty tank manually. Asset model is the use case of the intended function (i.e., Empty tank manually) includes (i.e., Turn valve)) wants to be performed by actor (i.e., Operator). Safety goal targets safety goal (i.e., System adaptation) represented with hexagon. In our example system boundary that is system elements has vulnerabilities that belongs to asset such as (i.e., No duplicate valve), (i.e., There exist no way to turn valve without motor), (i.e., No valve status indicator) that causes to system failure when it exploits and the harm (i.e., Can not empty tank) causes the system failure to be analyzed with risk modeling using FTA, harm element represented with snip diagonal corner rectangle.
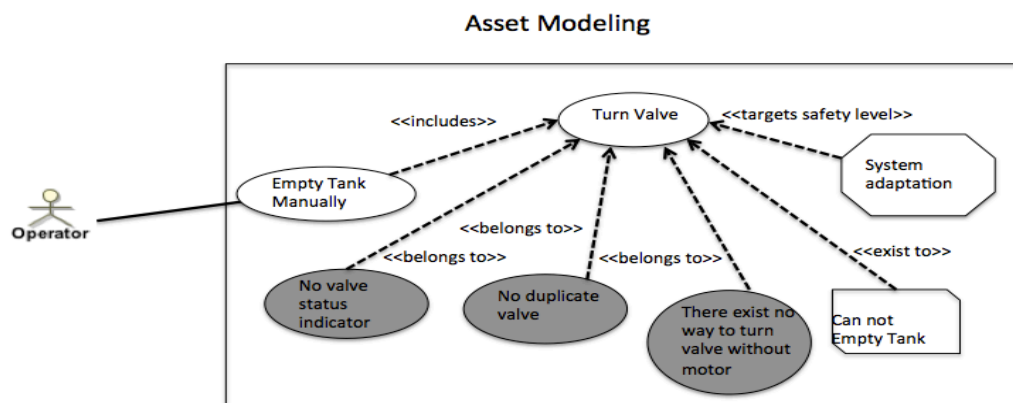
Fig 14. Asset Model

Risk Model

In Fig 15, we fault tree of present potential accidental scenario to the asset modeled with use case Fig. 15 regarding the intended function (i.e., Empty tank manually). Initial undesired event at the top of the tree is harm (i.e., Can not empty tank) asset gets. Intermediate event that is directly related to top event (i.e., Valve is stuck) is a hazard and also undeveloped event (i.e., Operator gives wrong command) is ate hazard that is also directly related to top event. And finally FTA completes with actual hazard that is an accident (i.e., Pressure too high)
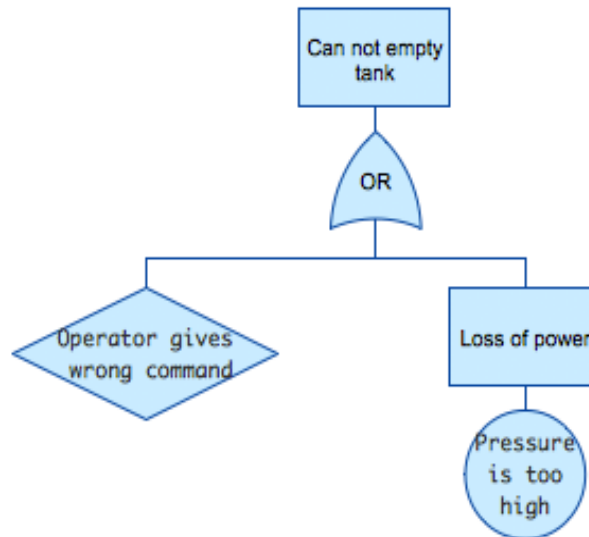
Fig 15. Risk Model

Safety Requirement Model

In Fig. 16, we present fault tree of present accidental scenario to the asset. The use case (i.e., Empty tank manually) includes safety requirement (i.e., Achieve turn valve without motor) represented as an attention symbol. Safety requirement eliminates the harm (i.e., Can not empty tank). And safety requirement ensures that tank valve will be able to open without motor and our elimination of safety risk will establish adaptation ability to our system that is safety goal.
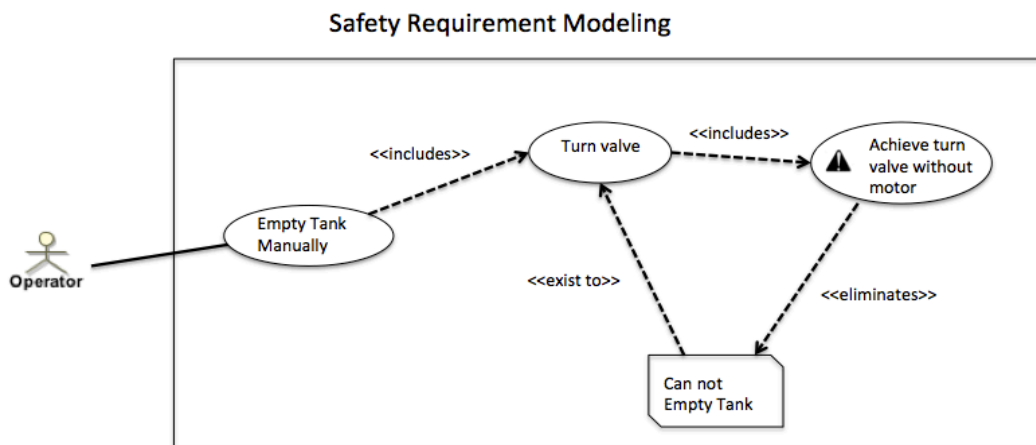
Fig 16. Safety Requirement Model

45

5.5 Results Comparison

Assuming that the risk analysis done by Stålhane for the boiler tank scenarios is correct, we compare our FTA risk analysis with it. Table 19 is presenting this comparison. In Table 19, first column shows SDM concepts; second column shows our safety risk analysis with two sub sections that are safety techniques we used as FTA and Misuse Case. Third column showing Stålhane's safety risk analysis with three sub columns first Synonyms defining similar concept meanings found in (Stålhane & Sindre, 2007) second and third are safety techniques FMEA and Misuse Case used.

From the Table 19, we see that our safety risk analysis with FTA and its alignment to SDM can elicit concepts as asset, accident, hazard, harm and safety requirements and they support with Stålhane's Safety Risk Analysis for the concept elements asset, accident, hazard, harm and safety requirement. This results support proves us correctness of our safety risk analysis with FTA.

Furthermore, our safety risk analysis with FTA and its alignment to SDM can elicit some other concepts elements as well such as safety risk, vulnerabilities and safety goal.

To sum up, from the comparison presented in the Table 19 and bullet points above it is clear that Stålhane's safety risk analysis supports our risk analysis with FTA and its alignment to SDM, which is validating the correctness of our method since we assume that results from Stålhane are correct.

Answers to validation questions:

Validation Question: Which safety risk analysis is better for the "boiler tank scenario" FTA and its alignment to SDM or FMEA?

Answer: Risk analyses done with FMEA and Misuse Cases for "Empty Tank Manually" use case supports with our risk analysis with FTA and its alignment to SDM. Moreover, FTA elicits some more concepts such as vulnerability, safety goal and safety risk. Regarding this answer coming from validation question, we could say that our alignment is correct and better in terms of SDM concepts it elicits.

5.6 Threats to validity

We follow some certain steps while performing FTA but at the same time we are aware of some conventions we used, so a random user would have some problems in understanding some of these and this may result with a wrong result or complications as mentioned below.

- When we do asset model with FTA, we include system component to the intended function wants to be performed, alignment is not prepared for purposes component not known for intended function.
- While doing risk modeling with FTA, it would be hard to predict all undeveloped hazards also some event names may remain unclear if user is not cautious about it.
- When doing safety requirement modeling user would have hard times in understanding that there needs to be drawn one safety requirement model for each accidental situation

such that if fault tree finds many accidents that results with same harm user will be in need of drawing many different safety requirements.

To sum up, if a random user of FTA and it's alignment to SDM does not recognize the following conventions we have while understanding the methodology we proposed then user may get wrong results or complications.

5.7 Summary

In order to control the correctness of our FTA and its alignment to SDM, we used a case scenario about boiler tank system and compared the results with Stålhane's safety risk analysis, which we assumed to be correct. In detail, for the use case "Empty tank manually" that is an intended function to be implement, we have made risk analysis and compared with analyses (models) from (Stålhane & Sindre, 2007). Results comparisons we have made acknowledged the correctness of FTA and it's alignment to SDM. Finally we have reported threats to validity. From overall case study, what we have understood is our methodology is correct when the system wants to be analyzed boundaries known. However, it is not tested in terms of ease of usage or performance as part of future work these are the factors to be improved.

SDM		Our Safety Risk Analysis (Section 7.2)		Stålhane's Safety Risk Analysis (Stålhane et al., pp. 425-426, 2007).	
		FTA	Use Case	FMEA	Use Case
Asset	Asset	-	Empty tank manually	-	Empty tank manually
	Safety goal	-	System adaptation	-	-
Risk	Safety risk	-Since the pressure is too high valve get stuck and operator can not empty the tank	-	-	-
	Accident	Pressure is too high	-	Pressure is too high	-
	Hazard	Valve is stuck	-	Valve is stuck	Valve is stuck
	Vulnerability	-	-No duplicate valve -There exist no way to turn valve without motor -No valve status indicator	-	-
	Harm	Can not empty tank	Can not empty tank	Can not empty tank	-
Risk treatment	Safety policy	-	-	-	-
	Safety requirement	-	Achieve turn valve without motor	-Valve status indicator -Duplicate valve -Must be possible to turn valve without motor	-
	Safety mechanism	-	-	-	-

Table 19. Comparison of Safety Risk Analyses for Boiler Tank System Scenario (Stâlhane & Sindre, 2007).

CHAPTER 6. CONCLUSION AND FUTURE WORK

We presented security and safety domains since we foresee that integration between security and safety starts in their domain models. And then, we presented extracted definitions for security and safety. Security domain (ISSRM) concepts definitions are taken from (Dubois et al., 2010). Safety domain concept definitions are collected and written by us. Then we have given sample to safety and security domain concepts and compared domain models in terms of their differences and similarities. Because we thought that similar concepts can be used for similar purposes. After this, we presented risk identification techniques from security and safety, as from security: misuse cases, mal-activity diagrams and secure Tropos and form safety: FTA, HAZOP, FMEA and PHA. We chose these techniques to present because these are categorized to be most frequently used risk identification techniques (Raspotnig, et al., 2013). Then, we presented step-by-step usage of FTA together with its ground rules and construction sample since we decided to choose FTA for the integration regarding these facts that its easy to use and understand also frequently used safety technique (Fault Tree Handbook, 1981).

After analyzing security and safety domain models and presenting the risk identification techniques they have, we realized the need of FTA and its alignment for the SDM. Thus, we have presented alignment of asset related, risk and risk treatment models of FTA for SDM. While making these alignments for asset model, we used misuse cases to represent the model because its aligned with ISSRM and since we think that some concepts from ISSRM are similar to safety such as asset, vulnerability, impact we thought of modeling asset with misuse cases is convenient. For risk model, we used FTA and elicit hazards, harm and accidents. Then, we have developed safety requirements for the risks we analyzed with FTA and modeled them with misuse cases.

To validate our work we have gathered a scenario (Stålhane et al., 2007) that is suitable and tested it for FTA and its alignment with SDM and then compared the results with Stålhane.

6.1 Limitations

In this study we are having limitations of the scope. This study focuses on two main titles first one is security and safety domain models along with their similarities & difference and second one is FTA and its alignment to SDM. The alignment of FTA with SDM includes interplay with Misuse Case in order to cover asset model and safety risk treatment model. However, we have not made an attempt to use a different security technique for modeling asset or creating safety risk model. Also we have not covered one of the usages of FTA, which is probability analysis of the risks that may arise and leave it out of our context.

We remain some levels for future research like carrying the interplay further by using a different security technique or creating an integrated technique between FTA and Misuse Cases in order to elicit security and safety problems alongside, in a way that considering safety as a security criterion such as confidentiality, integrity, availability and safety (Winther et al., 2001).

6.2 Conclusion

The idea for current thesis was inspired by two main research questions. First research question was similarities and differences between security and safety domain models and how can they be benefitted from each other. Based on the comparison made to capture similarities and differences between domain models (See Table 3), we have made a conclusion that security and safety asset

modeling can be done together and this could give us a chance to make security and safety analysis concurrently. And this exposed us our second research question, which is what kind of interplay can be done between security and safety techniques for security and safety analysis? In order to answer this research question we chose most suitable and frequently used (Raspotnig, et al., 2013) security and safety risk analysis techniques as Misuse Case and FTA. We claim Misuse Case and FTA to be the most suitable identification techniques to make interplay with because; firstly Misuse Case has alignment with ISSRM (Soomro, 2012). Secondly, FTA and Misuse Case are frequently used risk identification techniques (Raspotnig, et al., 2013). And finally, regarding the fact that FTA starts with initial undesired event that is harm to system and Misuse Case gives us chance to indicate harm in asset model to make risk analysis with FTA.

After deciding which techniques to make interplay with for security and safety analysis, we have decided to align FTA with SDM because security technique we made interplay with was aligned with ISSRM (Soomro, 2012) but safety technique was not and for interplay between techniques it was needed to understand capabilities and limitations of FTA with SDM. From the technique analyses we made for FTA on its usage, we realized that there was no way to indicate asset model and safety risk treatment model with FTA but risk analysis. And this exposed us to use security technique that is Misuse Case to cover asset model and safety risk treatment. By capturing risks with FTA, modeling asset and safety risk treatment with Misuse Case we have aligned FTA with SDM. We illustrated usage of our alignment with a running example (See Section 4.1).

In order to validate our work, we have controlled correctness of FTA and its alignment to SDM. We have gathered a scenario, and extracted safety risk analysis from this scenario then compared whether risk analysis extracted from the scenario support our risk analysis and identified threats to validity.

6.3 Future Work

When the overall thesis considered a specific future work could be realized, which is integration between Misuse Case and FTA. Because at the moment we have fundamentals for this integration since Misuse Case alignment exist with ISSRM (Soomro, 2012) and FTA alignment exist with SDM. Meaning that now a methodology can be created in order for further analysis of security and safety together regarding the fact that "The increased use of ICT-systems, in particular combined with the tendency to put "everything" on "the net", gives rise to serious concerns regarding security, not just in relation to confidentiality, integrity and availability (CIA), but also as a possible cause of safety problems" (Winther et al., 2001, p.1). Therefore we think that this could be a challenge and if the work is done it could be a sample for integration attempts of security and safety.

References

1. D.P. Eames and J. Moffett. The Integration of Safety and Security Requirements, Springer (1999)
2. Firesmith, D. G.: Common Concepts Underlying Safety, Security, and Survivability Engineering, Technical Note CMU/SEI-2003-TN-033 (2003)
3. Soomro, I. (2012) Alignment of Misuse Cases to ISSRM. University of Tartu, Estonia
4. Dubois, E., Heymans, P., Mayer, N., Matulevičius, R. A Systematic Approach to Define the Domain of Information System Security Risk Management. In: Intentional Perspectives on Information Systems Engineering. pp. 299-302. Springer (2013)
5. Mayer, N. (2009). Model-based Management of Information System Security Risk. University of Namur
6. Firesmith D.G. (2004). Engineering Safety Requirements, Safety Constraints, and Safety-Critical Requirements, ETH Zurich
7. Altuhhova, O. (2013). An Extension of Business Process Model and Notation for Security Risk Management. University of Tartu, Estonia
8. Raspotnig, C., & Opdahl, A. (2013). Comparing risk identification techniques for safety and security requirements, ScienceDirect
9. Mouratidis, H., & Giorgini, P. (2006). Secure Tropos: A Security-Oriented Extension of the Tropos Methodology. p.6
10. Altuhhova, O., Matulevičius, R., & Ahmed, N. (2013). Towards Definition of Secure Business Processes. University of Tartu, Estonia
11. Sindre, G. & Opdahl, A.L.: Eliciting security requirements with misuse cases. pp. 36-37. Springer (2004)
12. Winther, R., Johnsen, O.A., & Gran, B.A. (2001). Security Assessments of Safety Critical Systems Using HAZOPs, Springer (2001)
13. Brooke, P. J., & Paige, R.F. (2003). Fault trees for security system design and analysis. University of York, United Kingdom
14. Sindre, G. Mal-Activity Diagrams for Capturing Attacks on Business Processes. pp.2-6. Springer (2007)
15. Kletz, T. (1999). HAZOP and HAZAN. Davis Building, 165-189 Railway Terrace, Rugby, Warkwickshire / UK. Institution of Chemical Engineers
16. Braber, F., Hogganvik, I., Lund, M. S., Stølen, K., & Vraalsen, F. (2007). Model-based Security Analysis in Seven Steps—a Guided Tour to the CORAS Method. BT Technology Journal, vol. 25(1)
17. Ekelhart, A., Fenz, S., Neubauer, T.: AURUM: A Framework for Information Security Risk Management, Proceedings of the 42nd Hawaii International Conference on System Sciences (2009)
18. Asnar, Y., Giorgini, P., Mylopoulos, J.,: Goal-driven risk assessment in requirements engineering, Requirements Engineering, Springer (2011)
19. 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
20. Matulevičius, R., Mouratidis, H., Mayer, N., Dubois, E., Heymans, P.: Syntactic and Semantic Extensions to Secure Tropos to Support Security Risk Management, Journal of Universal Computer Science, vol. 18, 2004

21. Ericson, C.A. (2005). Hazard Analysis Techniques for System Safety. Preliminary Hazard Analysis (pp. 73-90). Fredericksburg/Virginia, John Wiley & Sons.
22. David, F. (1965). Advanced Concepts in Fault Tree Analysis. HAASL System Safety Engineering Missile Branch, Aero Space Division the Boeing Company. Seattle, Washington.
23. G. Sindre and A. L. Opdahl.: Template for Misuse Case Description. In Proceedings of the International Workshop Requirements Engineering: Foundation for Software Quality (REFSQ 2001), 2001
24. D. Kulak and E. Guiney, Use Cases: Requirements in Context, ACM Press, 2000.
25. Sindre G., Opdahl A.L.: Capturing Security Requirements through Misuse Cases, 2001.
26. US Nuclear Regulatory Commission, Fault Tree Handbook, NUREG-0492 (January 1981).
27. Marquis, H. (2008, November 26). Fault Tree Analysis Made Easy. Retrieved April 8, 2014, from http://www.itsmsolutions.com/newsletters/DITYvol4iss47.pdf.
28. Stålhane, T., Sindre, G.: A comparison of two approaches to safety analysis based on use cases. ER'07, Auckland, New Zealand. Springer LNCS, (2007)
29. Stamatis, D.H. (1995), Failure Mode and Effects Analysis – FMEA from Theory to Execution, ASQC Quality Press, New York, NY.
30. Nancy R.T. (2004). The Quality Toolbox, pp. 236–240, Second Edition, ASQ Quality Press.

Non-exclusive licence to reproduce thesis and make thesis public

I, **Servet Kurt** (date of birth: 02.03.1989),

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

Interplay of Misuse Case and Fault Tree Analysis for Security and Safety Analysis,

supervised by *Dr. 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**