jamk.fi

Security Risk and Threat Models for Health Care Product Development Processes

Aki Sivula

Master's thesis
November 2015

Master's Degree Programme in Information Technology
School of Technology and Transport

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

Description

Author(s)	Type of publication	Date
Sivula, Aki	Master's thesis	10.11.2015
	Number of pages 142	Language of publication English
		Permission for web publication: x

Title of publication
Security Risk and Threat Models for Health Care Product Development Processes

Degree programme
Master's Degree Programme in Information Technology

Tutor(s)
Huotari, Jouni
Kotikoski, Sampo

Assigned by
Tieto Healthcare & Welfare Oy

Abstract

The purpose of this study was to explore a variety of information security risk and threat models as well as apply and develop the appropriate model of health care needs of the domain. Healthcare applications are in transition, which stands for moving towards mobile, web and cloud applications.

Consequently, the security threats in the world have changed in health care sector with respect to harmful operators. For instance, the patient and personal data have become valuable assets on the local and international markets.

In principle, the modern application development team have to be able to respond to current threats and therefore plan and develop safe applications for health care area; however, before the actual application development the new type of security risk and threat analysis model is needed in order to take into account constantly renewing security perspectives during the application lifecycle with sufficient accuracy. The health care application development has a general need for information security risk and threat models, and the purpose of the thesis was to apply the existing models of Tieto's health care development model.

The research was successful, and it was implemented as case study research and the evaluation of the discovered models is based on literature analysis. Further evaluation was conducted after workshops with the Tieto staff members with a multiple choice questionnaire.

The results of the research were two threat models STRIDE and CAPEC for Tieto's Healthcare and Industrial Internet units. Additional result was an experimental theoretical attack library model based on behavior sciences which utilized antecedents of a person or a group, behavior and consequences.

Keywords/tags (subjects)
risk, threat, risk model, threat model, cyber security, information security, data protection

Miscellaneous
Attachments include workshop's PowerPoint presentations, the questionnaire and answers.

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

Kuvailulehti

Tekijä(t)	Julkaisun laji	Päivämäärä
Sivula, Aki	Opinnäytetyö	10.11.2015
	Sivumäärä 142	Julkaisun kieli Suomi
		Verkkojulkaisulupa myönnetty: x

Työn nimi
Security Risk and Threat Models for Health Care Product Development Processes

Koulutusohjelma
Master´s Degree Programme in Information Technology

Työn ohjaaja(t)
Huotari, Jouni
Kotikoski, Sampo

Toimeksiantaja(t)
Tieto Healthcare & Welfare Oy

Tiivistelmä

Opinnäytetyön tarkoituksena oli tutkia erilaisia tietoturvan riski- ja uhkamalleja sekä soveltaa ja kehittää niistä sopiva malli terveydenhuollon sovellusalueen tarpeisiin.

Terveydenhuollon sovellukset ovat murroksessa, jonka myötä ollaan siirtymässä mobiili- ja verkkosovelluksien sekä pilvipalveluiden maailmaan. Näin ollen tietoturvauhkien maailma on muuttunut terveydenhuollon sektorilla vahvasti haitallisten toimijoiden osalta, ja esimerkiksi potilas- sekä henkilötiedoista on tullut rikollisten keskuudessa arvotavaraa niin paikallisilla kuin kansainvälisillä markkinoilla. Lähtökohtaisesti nykyaikaisen sovelluskehitystiimin on pystyttävä suunnittelemaan ja kehittämään turvallisia sovelluksia terveydenhuollon käyttöön. Kuitenkin ennen varsinaista sovelluskehitystä tarvitaan uudenlainen tietoturvan riski- ja uhka-analyysimalli, jotta jatkuvasti uudistuvat tietoturvanäkökulmat voidaan ottaa sovelluksen elinkaaren aikana huomioon riittävällä tarkkuudella. Terveydenhuoltoalan sovelluskehityksessä on yleisesti tarvetta tietoturvan riski- ja uhkamalleille, ja työn aiheena on soveltaa olemassa olevia malleja Tiedon Healthcare-yksikön tuotekehitykseen joko uusien tai olemassa olevien sovellusten osalta sekä tarvittaessa jatkokehittää mallia tuotealueen tarpeisiin.

Tutkimuksen tulokset olivat hyviä ja työssä arvioitiin kirjallisuusanalyysin perustella riski- sekä uhkamalleja, joista kaksi valittiin käytännön testaukseen Tiedon ketterien tiimien työpajoihin. Jatkoarviointiin valittujen mallien arviointi toteutettiin työpajojen jälkeen vastattujen kyselylomakkeiden analysoinnin perusteella. Tutkimuksen tuloksena saatiin kaksi Tiedon Healthcaren sekä Industrial Internet-yksiköiden ketterään sovelluskehitykseen soveltuvaa mallia (STRIDE ja CAPEC). Lisäksi tuloksena saatiin kokeellinen käyttäytymistieteeseen perustuva teoreettinen hyökkäyskirjastomalli, jossa hyödynnetään aiempia yksilön tai ryhmän kokemuksia, käyttäytymistä sekä seuraamuksia.

Avainsanat (asiasanat)
riski, uhka, riskimalli, uhkamalli, kyberturvallisuus, tietosuoja, tietosuoja

Muut tiedot
Liitteenä työpajoissa käytetyt PowerPoint-esitykset sekä kyselylomake ja vastaukset.

Acknowledgements

The thesis required a great deal more time than expected and even caused some frustration before it was started. A few years ago I was searching for cyber security testing related issues; however, the thesis made me enter the world of threats. I would like to thank Piotr Krawczyk for a course that really opened my eyes. I learnt very quickly that it is not important to get any topic for thesis as it will lead to only average results but instead of that, follow your passion. The original topic selection failed after 100 hours of investigation work; however, eventually it helped a great deal to get me interested and finally I made the correct decision after two failed thesis topics. Steve Jobs said once "Go out and a get a job as a busboy or something until you find something you're really passionate about." And that is true – life is too short to be wasted doing boring work. Special thanks also go to Heikki-Pekka Noronen who offered the topic for me and made the thesis possible. Special thanks also go to Tieto Healthcare & Welfare for permitting the use of their resources and of course, special thanks go to my family for supporting my studies that were mostly carried out outside working hours.

ACRONYMS AND ABBREVIATIONS

Term	Description
ABA	Applied Behaviour Analysis
APT	Advanced Persistent Threat
CAPEC	Common Attack Pattern Enumeration and Classification
CEL	Common Exposure Library
CIA	Confidentiality, Integrity, Availability
CISSP	Certified Information Systems Security Professional
DESIST	Dispute, Elevation of privilege, Spoofing, Information Disclosure, Service Denial and Tampering
DREAD	Damage, Reproducibility, Exploitability, Affected users, Discoverability
JIRA	Issue and project tracking software by Atlassian
MOL	Methods and objectives library
OWASP	The Open Web Application Security Project
P.A.S.T.A.	Process for Attack Simulation and Threat Analysis
SDL	Software Development Lifecycle
STRIDE	Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, Elevation of privilege
TAL	Threat agent library
TARA	Threat Agent Risk Assessment
TRIKE	Conceptual framework for security auditing and threat modelling from point of risk management view.

CONTENTS

FIGURES

TABLES

1 Introduction and thesis objectives

Risk and threat models are cornerstones of software development security. Basically, their feasibility has been proven on operating systems development; however, now they are coming into wider usage on tailored applications and systems. The idea of this thesis is to explore risk and security models that have helped companies to locate flaws in software architecture and software and eventually select or develop a model for health care software development. Moreover, the thesis can give a direction and new methods for software development teams and management.

1.1 Background for Problem

Health care has been traditionally a safe harbour from the Internet's and other today's network threats since it has been separated from public networks; however, the world is now changing much faster than few decades before. Basically, the most important physics laws and secrets have been revealed; however, networking and social networks are going through a continuous and huge change. Social networks might be in key position for future society even though they are considered entertainment instead of serious communication media.

For example, over 20 years ago everyone got most of their news and new data from TV news and newspapers. The information was often hours or even days old. The Internet era changed data delivery very quickly and the first home pages emerged. Social networking services and smartphones with cameras added even more speed to information delivery. It is a benefit; however, it is also a threat since basically nearly any data can be stolen and shared in real-time. For example, sharing celebrities' personal life data and photos is popular fun for a large group of people.

Moreover, people tend to be curious to get personal health data, particularly when speaking about relatives, a former spouse or even a famous person. One well-known example in 2013 was Michael Schumacher's skiing accident in December 2013 (BBC Sport, 2013). Eventually the rumours flew concerning Schumacher's health. The case

was an interesting target for tabloid press and individuals. It was obvious that some people filmed him at hospital, and stealing or getting personal health data was tempting as well. As a proof of rumours and general interest in health data in March 2015, Schumacher doctor's laptop was stolen (Narciso, 2015).

Based on an article written in Wired magazine article, security researchers examined the security of hospital networks and found many of them "leaking valuable information to the internet, leaving critical systems and equipment vulnerable to hacking" (Zetter, 2009). The article also mentions that the problem was an unpatched computer with an Internet connection. Exploiting was easy without user interaction and after that the road was open to medical devices that the user wanted to attack (Zetter, 2009). One shortcoming of health care organizations was that they do not take security seriously and security flaws exist in most of the health care devices. The article mentioned flaws that were find in defibrillators and infusion pumps, and the attacker could simply manipulate dosages or give shocks to a patient's heart remotely.

Zetter lists systems with were exposed as follows:
"Among the systems with exposed data, the researchers easily identified at least 32 pacemaker systems in the organization, 21 anesthesiology systems, 488 cardiology systems, and 323 PACS systems—radiology systems for reading X-Rays and other images. They also identified telemetry systems, high-risk systems that are often used in infant-abduction prevention systems as well as for monitoring the movement of elderly patients throughout a hospital to ensure they do not wander off." (Zetter, 2009).

One example of an internal threat was a case in Jyväskylä where city health care employee snooped over 100 patients' data without any kind of permission or reason. The follow-up of the patient data was done with random log audits (Doagu, 2012). Based on the previous article such cases are rare, and in particular, when snooping a wide group of people. Since the entire log data is not checked for all employees with audits, there is still a possibility that someone might snoop data without ever getting caught, especially if snooping is carried out in a deceptive way for a selected target.

1.2 Requirement for Development

Security has always been an important issue on health care area and will remain so, however, the threats and risks are continuously changing. As mentioned in chapter 1.1, many medical equipment contained security vulnerabilities. The basis of this thesis was to study the security area and particularly the risk and threat analysis models, and test whether they apply to software development within health care. The objective was to mitigate the previously mentioned risks and threats. Additionally, the aim was to develop a new generic model combining the best parts of the existing models, and that suits the health care area in particular.

1.3 Research Problem

Tieto Health care needs a new risk and threat model as an approach for long term software development and maintenance. With the model software development teams and management should overcome security based risks and threats in the future. Eventually the goal is to develop a new model based on the best parts of existing models. The selected models are tested with real software teams on Tieto's healthcare & welfare organization. Tieto's healthcare unit requires that the model is easy to use and the usage of the model should not consume a great deal of time. The model should reveal the threats in effective way without too strict formality. Too formal or very complicated model would not be applicable in long term usage since teams are using the agile methods.

1.4 Research Questions

The research problem is basically generic for all organizations that develop software for health care area. Based on that, the following primary research questions were recognized.

1. What appropriate risk and security models already exist?
2. Which existing risk and security model fits best for health care software development processes?

3. What are the weaknesses of existing models?
4. Does the developed health care specific risk and threat model offer better risk and security management than generic models?

1.5 Research Plan

The following steps are included in this thesis as follows:

- Research what risk and security models are available.
- Investigate health care software area security issues.
- Research theory based models based on literature analysis.
- Select 2 – 3 threat models for practical test for the development team(s)
- Give guidance regarding to the threat model for the development team
- The development team applies the threat model to the daily software development work
- The development team opinions are collected within an online questionnaire as primary data collection method.
- Documentary data as secondary method to find out threat model advantages and disadvantages from the news archives, the information security theses and the security books.
- Estimate feasibility of models and possible changes required for them.
- Evaluate the risk and threat models based on collected data and literature review.

1.6 Research Method

The selection of the research method started with the investigation of the research onion model (Saunders, Lewis & Thornhill, 2012, 128). This philosophy was selected due to the demand to apply invented or found methods to support product lines and their projects in a relevant way. The research questions can be considered unambiguous; however, they do not define the particular philosophy thus giving a possibility to use different philosophical viewpoints if needed (Saunders, Lewis & Thornhill, 2012, 130). The selected research approach is abduction since a possibility to move

from theory to data and vice versa. Eventually the idea was to apply existing models to get data but also tuning the selected models based on collected data. (Saunders, Lewis & Thornhill, 2012, 130). Methodological choice selected was multi-method qualitative study and in this thesis it intended to be done with interviews and shadowing software development teams at work. (Saunders, Lewis & Thornhill, 2012, 165). However; shadowing was not possible from Tieto's side due to too long shadowing time. Estimated shadowing time was minimum 2 weeks which means one development sprint length. Eventually it was decided to select literature analysis as the primary method and a self-completed questionnaire as a secondary method so the model evaluation is possible with the one or more development teams. The selected strategy for the research is case study and ideally it is a good way to explore a research topic in product lines and projects of real software development. The chosen time horizon selected was cross-sectional. The longitudinal approach would give better results in this case.

Table 1. Selected research philosophies and approaches

Research philosophies and approaches	**Selection**
Philosophy	Pragmatism
Approach	Abduction
Methodological choice	Multi-method qualitative study (The primary method is literature analysis, the secondary method is a self-completed questionnaire)
Strategy	Case study
Time horizon	Cross-sectional.
Research design	Case study

2 Health care specific Security risks

PwC reports yearly the security snapshot for each industry area. The cyber security challenges report 2014 on the field of healthcare industry describes top 5 security challenges in 2014 and outlook to the security area. The data is collected by PwC

from “The Global State of Information Security Survey 2015”. As seen on Figure 1 the access control and identity management for end users are clearly on highest place. The data leakage, cloud computing, encryption in storage and in transit and regulatory requirements are following with high percentage. Some of the new advanced technologies mentioned in the report are Internet, telemedicine, mobile devices, social media, information sharing and Big Data analytics. The report underlines that technology transforms healthcare payers and providers’ interaction with their patients, business partners, and regulators. (PwC, 2014).

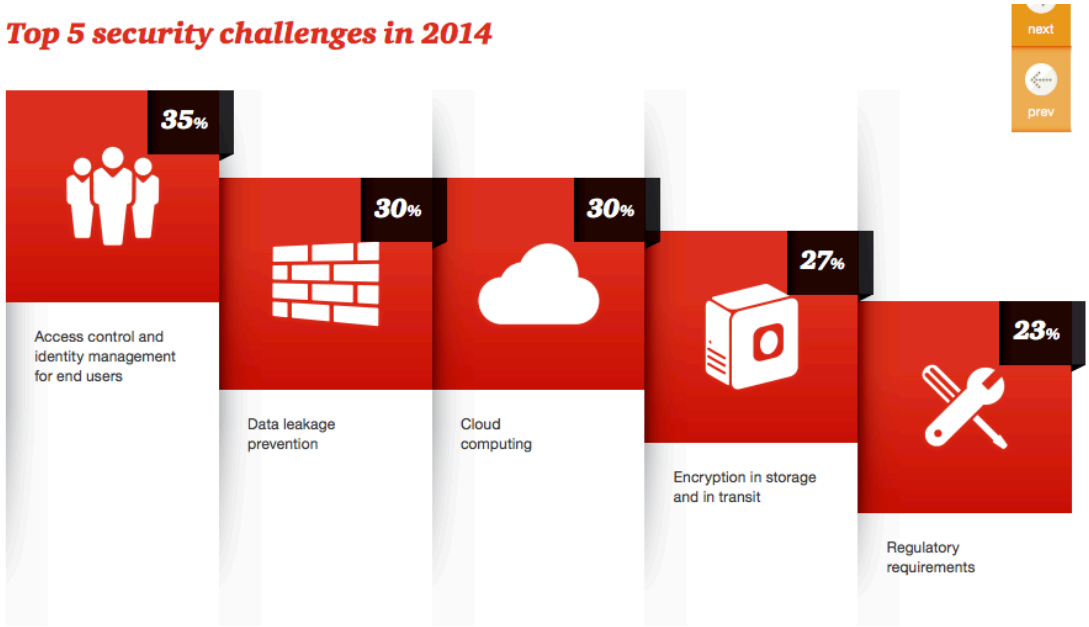

Figure 1. “Top 5 security challenges in 2014“. (PwC, 2014).

The sources of the incidents are definitely worth of investigation to understand what the most likely human based attacking vector is that should be prevented or mitigated. According to PwC, current employees are still the highest risk to cause new security incidents; however, also former employees and hackers are common threats. The threat of foreign nation-states has grown from 2% to 5%, and thus it is a fast growing threat that must be taken into account (PwC, 2014).

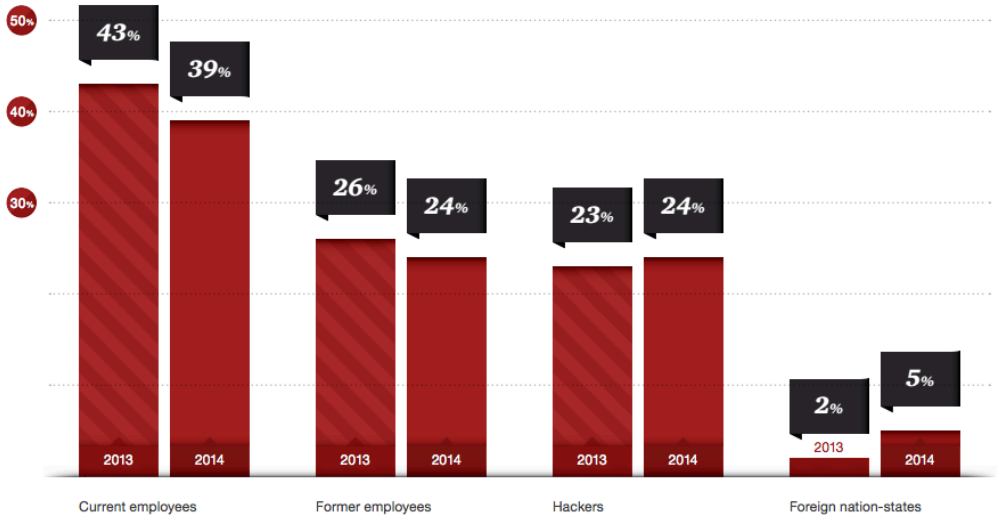

Figure 2. “Sources of Incidents” (PwC, 2014)

According to PwC, “Medical records are more valuable because cybercriminals can use them to create an identity, as well as carry out sophisticated insurance fraud schemes” (PwC, 2014). Based on the report, the implementation of the electronic health records is the highest risk; however, also data sharing in different forms plays a very significant role. Careless data sharing might lead to unwanted data disclosure.

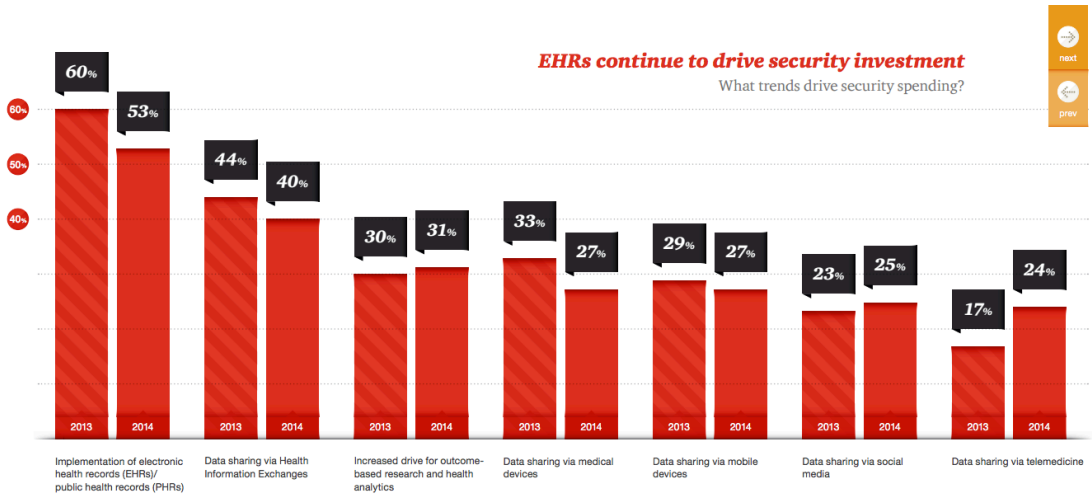

Figure 3. “EHRs continue to drive security investment” (PwC, 2014)

As a conclusion, PcW report on internal threats, software implementation, data sharing, new technology and top security threats shows the roadmap for future; however, it is not limited to them. The selected threat model should meet today’s security

requirements to prevent or mitigate the security risks; however, the threat model should be flexible and easy to apply for future threats or other threats that are not listed on the security report.

3 Threat Models

This part of the document describes the theory of threat models. Several different threat models and threat modelling approaches are described.

3.1 Threat models theory

Threat models can be considered as any models the purpose of which is to help to solve and overcome certain problems. The simplified threat model is basically a high level description of how the attacker could exploit the possible vulnerabilities on the application environment (Velez & Morana 2015, 1-3). According to (Velez & Morana 2015, 1); the generic application threat model definition includes strategic process, attack scenarios, vulnerabilities, application environment and identification of the risk and impact levels. Software development security and war strategies share many similarities as both have malicious attackers and defeating of enemy requires always having a better strategy than the enemy.

Harris described the threat as follows: “A threat is any potential danger that is associated with the exploitation of a vulnerability. The threat is that someone, or something, will identify a specific vulnerability and use it against the company or individual”. (Harris, S. 2014). According to Shostack (2014, 3), threat modelling is simply way to find security problems beforehand and see the bigger picture instead of the code.

For example, if software is developed without taking threats into account the software might be delivered to a customer and everything seems to be fine for a long time. A bunch of normal bugs might be fixed, however, there is no feedback of security flaws lurking in the system. A disaster may follow quickly after a malicious at-

tacker attacks against the system. To overcome threats they must be recognized and handled properly. Overall recognition of the threats might be easier if war strategies and generally strategies are familiar but it is not required at all. The next paragraph discusses a fictional account as an example of losing reputation due to multiple vulnerabilities. (Velez & Morana 2015, 1-3)

3.1.1 Threat example

An example system handles X-ray images; however, it is running on an obsolete operating system with very low or no defence against intruders. The system works well, the customer is satisfied and business continues without problems. Basically everything works fine many years without serious problems. One day the hospital takes care of a famous person after a serious accident. Meanwhile, youngsters have found that one of the hospital computers has access to nearly everywhere in the hospital network. Tabloid press looks for news about the celebrity and the youngsters knows that very well. A computer that contains X-ray images is hidden somewhere in the hospital network; however, the attackers have done few days of reconnaissance and found a computer with many other interesting targets.

Because the system itself does not have any defence or strong authentication mechanisms, collecting images is very easy. After a few days the X-ray images of the celebrity are on tabloid press and the attackers got some extra money. A lawyer contacts the hospital and soon the hospital contacts a software developer. The hospital is found guilty for abandoning security of the networks and the tabloid press nails the software developer down and causes severe damage for business because in a short time an important contract is lost for competitor due to better security. Luckily, noone died because of the mistake because soon a new security researcher of the hospital found a security flaw in a Bluetooth-enabled defibrillator that can be manipulated to deliver shocks to a patient's heart.

3.1.2 Strategy against threats and the security concept

The following citation acts as a guiding principle in this study “Victorious warriors win first and then go to war, while defeated warriors go to war first and then seek to win” (Tzu, S. 1913). Without proper modelling and planning of software the team develops first and then seeks to win after committing Himalayan blunder. Definitely this does not stand for endless amount of planning and hundred pages specifications but more like a continuous strategy follow up with situation awareness and continuous thinking in the changing environment and world.

The following figure is from CISSP Exam guide (Harris, 2014, 27) and explains clearly a threat position on a different security context. As seen the model is asset based; however, threats can be handled. In the previous fictional story youngsters were threat agents and threat was an exploitation of missing or weak authentication bypassing. Figure 4 describes the chain of the threat agent, threat, vulnerability, risk, asset, exposure and the safeguard. Understanding of the chain helps to understand the role of the threat in big picture. However; it is worth to remember that evaluation of the threat risk can help to make decision whether the impact is serious enough. All threats and risks are not worth to prevent or mitigate.

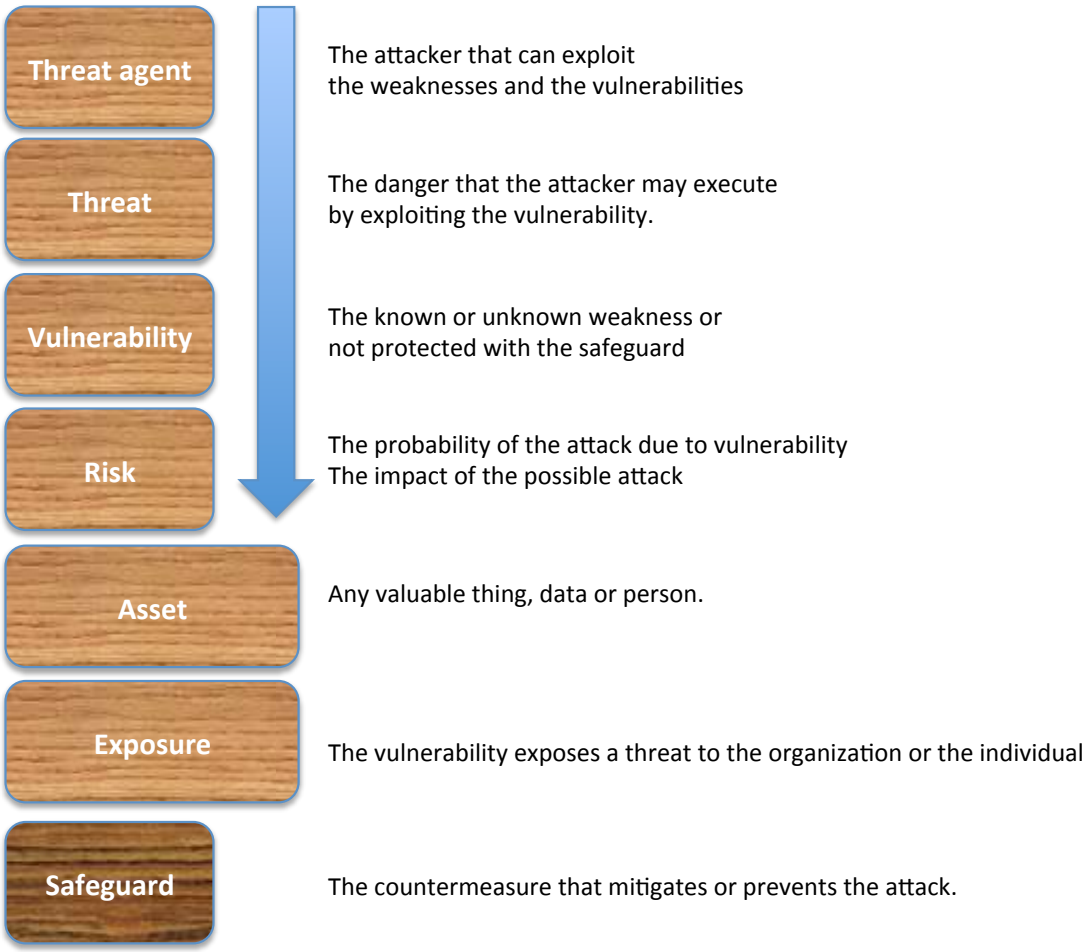

Figure 4. The security concepts and their relationships (adapted from Harris, 2014, 27)

3.2 DREAD – Risk Assessment Model

DREAD stands for Damage, Reproducibility, Exploitability, Affected users, Discoverability. DREAD uses categories to assess the rating of a threat. Three is the highest rating, two medium, one low and 0 none. The ratings can be summarized for given exploit to prioritize exploit. It was Microsoft's attempt to improve security with risk modelling during software development. There was an article of it in MSDN 2003 but support for DREAD deceased 2010 (Shostack, 2014, 180). However; next chapter includes short description how the risk calculation works by the model.

Table 2. DREAD meaning (Adapted from OWASP, 2015, Risk Threat modelling)

Word	**Description**
Damage	Damage stands simply for the seriousness of an attack.
Reproducibility	Is the attack repeatable and how easy would it be to repeat the attack?
Exploitability	Exploitability means ease of the attack.
Affected users	Affected users stands for all people impacted by an attack
Discoverability	Discoverability means how easy is it to discover the exploit.

3.2.1 DREAD usage

Calculating risks with DREAD is based on risk factors. Each term gets a value and the sum of the values is divided by constant 5 (amount of terms). The result forms the DREAD risk factor. The higher the factor is the higher the risk and vice versa. (OWASP, 2015, Risk Threat modelling).

Following chapter describes short instructions for the quantifying the DREAD risks (OWASP, 2015, Risk Threat modelling):

- Each risk gets a value between 0 – 10. Number 0 is lowest and 10 highest.
- Quantify the each category with questions. Quantifying depends on the system, the participants and planning of the questions.
- Review and iterate to find best set of the questions

Table 3. DREAD example questions (Adapted from OWASP, 2015, Risk Threat modelling)

Example questions	**Example answers**
How much damage does the used exploit cause?	0 = No damage 5 = Medium damage. The error message and the user name with address re-

	vealed. 10 = Disastrous damage, all patient information available.
Is it possible to reproduce the attack?	0 = Very difficult or impossible 5 = Gain administrators account and run it. 10 = Run it on the browser without user account.
How easy is it to attack? What is needed?	0 = Not possible 5 = Login to the application with the administrator account needed 10 = Anyone can run it just on the browser address line
Who is affected by the attack?	0 = Nobody 5 = System users (administrators, users) 10 = Company users and company website users.
How easy is it to find the exploit?	0 = Nearly impossible 5 = Developer with good hacking skills can find it. 10 = Just click the buttons in different order

3.3 Brainstorming as an threat analysis method

Brainstorming consists of two parts, the first phase being the generation of new ideas and the second phase the analysis of the ideas. The goal is to collect all ideas in the first phase without judgment. (Shostack, 2014, 31-34).

3.3.1 Brainstorming methods

As a method brainstorming is easy to arrange; however, its results might be difficult to address. Basically brainstorming may give interesting viewpoints for risk and

threat analysis; nevertheless, it works only as supportive action since threats might be completely out of team responsibility area. For instance, hardware related threats might be interesting for someone in user interface development team, however, addressing them is a completely different story. Typically recognized threats heavily rely on participants' experience and time used for the brainstorming meeting. Additionally, facilitation of the meeting is essential in order to move on. The team also needs a technical expert of the area so that the evaluation of the ideas is more successful, and more used time for brainstorming stands for a higher quality of results. Despite facilitation or usage of time, the results are mostly useless whenever participants are not interested in brainstorming or facilitation is badly organized. Brainstorming requires removal of boundaries and scope so that the threats may be difficult or impossible to prove. The second problem is to define the exit criteria for the brainstorming since boundaries and scope are removed (Shostack, 2014, 31-34).

3.3.2 Scenario analysis

Scenario analysis stands for written scenarios and the purpose of the analysis is to ask what may go wrong in the scenario. As an example, a Hell's Angels member could get a bike and the donor could think what that person could do with the bike: drive to church, go to blackmail someone, sell it without the donor's knowledge and so forth. (Shostack, 2014, 32).

3.3.3 Pre-mortem

Pre-mortem stands for gathering of professionals and assignment to find out beforehand what went wrong after the project or an important milestone. Basically, it is just an imaginary way to find out why a project or milestone failed before the real failure even happened. Eventually, the idea is to explore why participants think that a product will fail from the point of view of threat or risk. (Shostack, 2014, 33).

3.3.4 Movie plotting

In movie plotting brainstorming is moved under movie style theme. Attendees will pick one or more movies and they can for example imagine themselves as agents who want to intercept a government computer through the Internet. The more provocative or outrageous the ideas are the more they should help to generate new ideas. Eventually trying a different style of movies and roles may reveal different styles of threats.

Movie plotting is more entertaining than an effective way to recognize attack ideas. The purpose of the method is to throw fuel on fire so that the flow of ideas will be achieved in full scale. (Shostack, 2014, 33).

3.3.5 Literature review

Literature review is always a part of a thesis; however, it is also a brainstorming method. Basically, similar methods that are applicable for thesis literature review are applicable for literature review brainstorming. For instance, different kinds of search engines, academic literature or theses can reveal many new ideas. Also, searching According to domestic and foreign competitors' products related threat or risk information might be helpful. (Shostack, 2014, 33).

3.4 Microsoft Threat Modelling Methodology

Microsoft has its own Threat Modelling technology and its development started with a directive in January 2002. Actually the design phase threat modelling technology is a part of a bigger picture. The technology was developed to find ways to improve the existing security code. Since malicious software rose after the millennium, Microsoft was forced to improve the software security and they have continued to improve it since then. (Microsoft, 2014).

The directive was written by Microsoft's Trustworthy Computing team and the final result was Microsoft Security Development Lifecycle. The policy has been mandatory

in Microsoft since 2004, and it is part of the software development process (Microsoft, 2014). After one year usage of SDL for Windows XP and Vista vulnerabilities were 45% less than previous year before SDL (Microsoft Security Development Lifecycle Core Training Classes, 2010).

Basically methodology is strongly directed for Microsoft based product development; however, not limited to. Fundamental principles of all information security are confidentiality, integrity and availability (CIA) supported by authorization, authentication and non-repudiation (Harris, S. 2014, 22-24). Microsoft's model uses tailored methodology to meet three CIA basic tenets' requirements and its acronym is "STRIDE" (Spoofing, Tampering, Repudiation, Information disclosure, Denial of Service and Elevation of Privilege). (Shostack, 2014, 66). Basically such tailoring gives easier concepts to follow and strongly supports confidentiality, integrity and availability basic principles. The former threat model before STRIDE used by Microsoft was DREAD. (Shostack, 2014, 180).

The SDL contains wide range of threat material like SDL Process Guidance, SDL Optimization Model, SDL Pro Network, SDL Threat Modelling Tool and SDL Process Templates. Their original upper level process contained process phases Training, Requirements, Design, Implementation, Verification, Release and Response. Each phase had specific requirements and the threat modelling is part of the design phase. The one part of security is handled with the threat modelling in design phase. After the design phase all features and the product architecture are reviewed and the threats and mitigations are recognized. (Microsoft Security Development Lifecycle Core Training Classes, 2010).

3.4.1 SDL Threat Modelling Process and the tool

The following figure describes the current Microsoft SDL thread modelling process from a point of planning view with the tool. It is possible to use the process on the whiteboard, however, Microsoft has released a free tool for drawing threat models for the software. Since Microsoft uses STRIDE as a threat model the tool has direct integration for STRIDE per each element. The tool is targeted for developers, thus no

security background is required. Additionally, SDL Threat Modelling Tool contains support for issue tracking systems and contains reporting capabilities. (SDL Threat Modelling Tool, 2015).

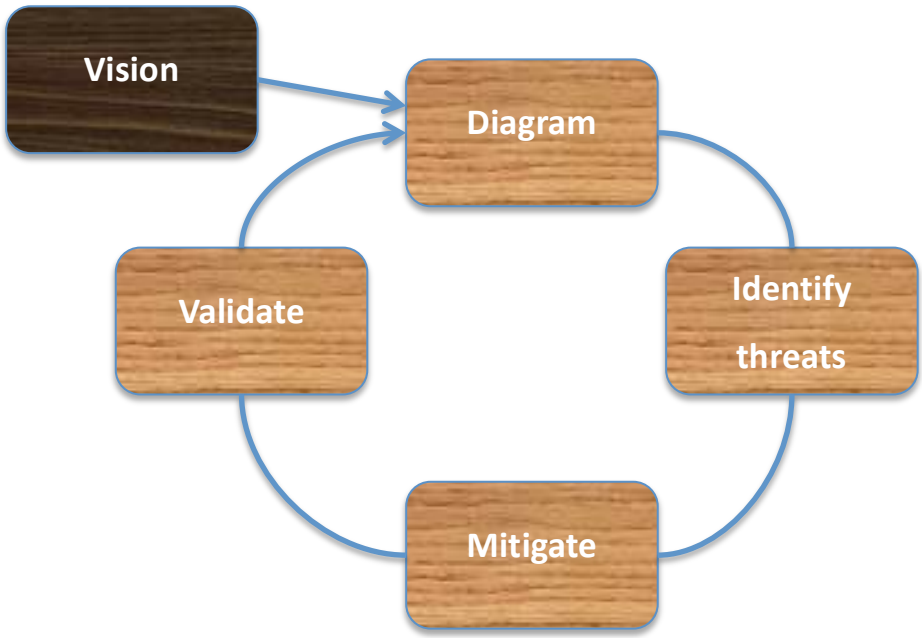

Figure 5. SDL Threat Modelling Process (adapted from Microsoft SDL Team, 2015)

The methodology is built to use the graphical data flow concept that includes a standard set of graphical symbols. Typically, modelling is simply a concept that can be applied to reality in many different ways. Basically, Microsoft Threat Modelling Methodology can be applied without any tool, however, Microsoft's strategy has been to offer tools with methods so customers will support their ecosystem. Nothing prevents from using the tool for any other product area as symbols are mostly generic. However, many stencil names are Windows branded and all other operating systems are completely missing. User can rename components' captions as wanted; however, the original properties stay as constant. In other words, Windows runtime caption can be anything; however, the property name "Windows runtime" is constant.

As a generic note, the tool usability is also pleasant and in line with other Microsoft tools. Unfortunately, the file format on Threat Modelling Tool 2014 is completely new, and moving the model to other tools might be a barrier in certain cases due to the different Office tool versions. (SDL Team, 2014).

The modelling style is aligned with other Microsoft tools like Visio and the model uses graphical Data Flow Diagrams that have a standard set of symbols. The symbols contain data flows, data stores, processes and interactors. (SDL Team, 2014).

Advantages of the tool are listed below as follows:

- Simple (Shostack, 2010)
- Standard set of symbols (SDL Team, 2014).
- Supports STRIDE with guided analysis and mitigations (Shostack, 2010)
- Bug tracking tools integration support (SDL Team, 2014).
- Office tools support (also Visio) (Shostack, 2010)

Disadvantages of the tool are presented as follows:

- The tool runs only on Windows platform (SDL Team, 2014).
- Certain stencils are for Windows based systems only (The author)

3.5 STRIDE

Acronym STRIDE is stands for the words spoofing, tampering, repudiation, information disclosure, denial of service and elevation of privilege (Shostack, 2014, 61). The threat model was invented and produced by Loren Kohnfelder and Praerit Garg in 1999 in Microsoft, and the first entry was on MSDN Blogs on the first of April 1999 (Shostack, 2014, 61). According to Kohnfelder and Garg (1999, 1), Microsoft should use STRIDE to identify threats during the design phase. In principle each word works as an umbrella for number of the similar style of threats. For instance spoofing is possible for the IP or MAC addresses on the networks but as well for a fake user profile that was created in deceptive purposes. The primary purpose of the STRIDE is to recognize the possible threats and gather the possible attacks or threats. The categorization is secondary issue and often users have to select one of multiple choices and do not worry about the right category (Shostack, 2014, 64).

Despite the STRIDE being a lightweight threat model, the usage requires a vast amount of work. (Shostack, 2014, 64). The easy to understand approaches are more

useful for beginners; however, after gaining experience more demanding models may be more useful. (Shostack, 2014, 409-411). The threat prevention is based on investigation of the right defences for the possible attacks on the target system.

STRIDE is planned to prevent the threats; however, the vulnerabilities and the management of the vulnerabilities coverage are not throughout covered (Shostack, 2014, 221). The other variants are STRIDE-per-Element and STRIDE-per-Interaction. (Shostack, 2014, 80). The variants work with the same ideology; however, they do require a deeper understanding of the STRIDE to get the best results. (Shostack, 2014, 61-62).

Table 4. STRIDE threats and violations (adapted from Shostack, 2014, 62)

Threat	**Property violation**
Spoofing	Authentication
Tampering	Integrity
Repudiation	Non-repudiation
Information disclosure	Confidentiality
Denial of service	Availability
Elevation of privilege	Authorization

3.5.1 STRIDE usage

STRIDE as a model fits to the many different development models and processes and it is possible to apply to all kinds of development models. One approach is to use the whiteboard and draw everything with the team. An official possibility is to use Microsoft's Threat Modelling Tool and draw the system's interfaces and components as a diagram. Applying the STRIDE to the diagram can be done in a formal way with design work; however; the problem with the tool usage causes often less collaboration and the one person perspective is taken into account. It can also be a tedious task and often only security experts participate in review meetings. The second and more entertaining possibility is to use Elevation of Privilege game. The game is created

based on Protection Poker by Laurie Williams, NCSU and it is promised to offer an easy way for developers to enter to the world of threat models'. The game requires the diagram of the system with its interfaces or the selected part of the system. During the game the threats in the cards are connected to the diagram under investigation. In the end the points are counted, a winner is declared and all found threats will be reported as bugs. (Shostack, 2010).

3.5.2 Spoofing

Spoofing stands for masquerading the target to something else by tampering the target or the data. Examples of spoofing targets are processes, computers or persons.

Spoofing can be done for instance for the process, the file, the computer or the person. For instance, computer spoofing is possible to apply for ARP, IP or DNS services. The person spoofing has as well different possibilities like spoofing the name on the email or hijacking the user account. (Shostack, 2014, 64 - 66).

3.5.3 Tampering

"Tampering is modifying something, typically on disk, on a network, or in memory". (Shostack, 2014, 67). Tampering on the computer includes modifying any file on the disk or memory. Tampering on the network includes adding, modifying or removing the packets. (Shostack, 2014, 67).

3.5.4 Repudiation

"Repudiation is claiming you didn't do something, or were not responsible for what happened". (Shostack, 2014, 68). The repudiation can happen due to accident or due to purpose. Repudiation may be very difficult to prove if the logs are missing, not available or made unreadable by any reason. (Shostack, 2014, 68-69).

3.5.5 Information disclosure

“Information disclosure is about allowing people to see information they are not authorized to see”. (Shostack, 2014, 70). Information disclosure can happen against a process, data stores or data flow. (Shostack, 2014, 70-71).

3.5.6 Denial of Service

Denial of service stands for consuming all resources and therefore preventing usage of a provided service by malicious means. The denial of service can be done against process, data store or data flow (Shostack, 2014, 72).

3.5.7 Elevation of privilege

Elevation of privilege stands for allowing the attacker to do something that is authorized for higher level user accounts. For instance, the elevation of privilege can be done with corrupting the process, passing the authorization with missing authorization checks or with data tampering. (Shostack, 2014, 73).

3.5.8 Exit criteria

There are two different exit criteria for the normal STRIDE variant. Easiest is to have one threat per threat type. Second and more comprehensive possibility is to have diagram that has one threat per element. The STRIDE variant STRIDE-per-element has most comprehensive exit criteria and it stands for one threat per check. (Shostack, 2014, 85).

3.6 DESIST

DESIST is very similar with STRIDE, however, the repudiation is replaced with dispute, and denial of service is simply service denial. An acronym is built of Dispute, Elevation of privilege, Spoofing, Information Disclosure, Service Denial and Tampering. (Shostack 2014, 85). Unfortunately only one reference was found concerning DESIST and therefore it cannot be described in more deeper manner.

3.7 P.A.S.T.A (Process for Attack Simulation and Threat Analysis)

The security area evolves all the time and PASTA is attempt to answer for new sophisticated cyber threats with its integration to the companies' existing security engineering, risk management, incident response and vulnerability management processes. The process should especially help to develop systems with resilience to the targeted attacks on web sites and services, e.g. distributed denial of service attacks or malware attacks. (Morana, 2014, 2).

Regarding to Morana (2014), the main goals of the PASTA process are improving visibility of cyber threat risks, extending the organization protection domains, leveraging existing application security processes, integrating with the SLDC and increasing the maturity. The mentioned benefits are risk reduction, knowledge of the threats, resilience, software security and collaboration among stakeholders. The stakeholders are developers, security or pen-testers, project managers, architects, CISO's and business managers. (Morana, 2014, 4).

The model has seven different stages and each stage steps are enumerated with prefix of the stage. In first stage all business objectives, security or compliance objectives, impact and the risk profile are handled in appropriate manner. The second stage concentrates on software, counterparts, services, infrastructure and the technical scope completeness. The third stage handles use cases and risk functions, document data flow diagrams (DFDs) and functional and architectural decomposition analysis. The fourth stage is the threat analysis part including scenarios, threat information, threat libraries, threat agents and the probability of the threats. The fifth stage is reviewing and identifying the weaknesses and vulnerabilities. The sixth stage is attack strategy, attack modelling, testing and simulation. The seventh and final stage is risk calculation, countermeasures identification, risk mitigation, residual risk calculation and recommended strategies for risk mitigation. (Morana, 2014). The following figure illustrates PASTA's main stages and the stage contents.

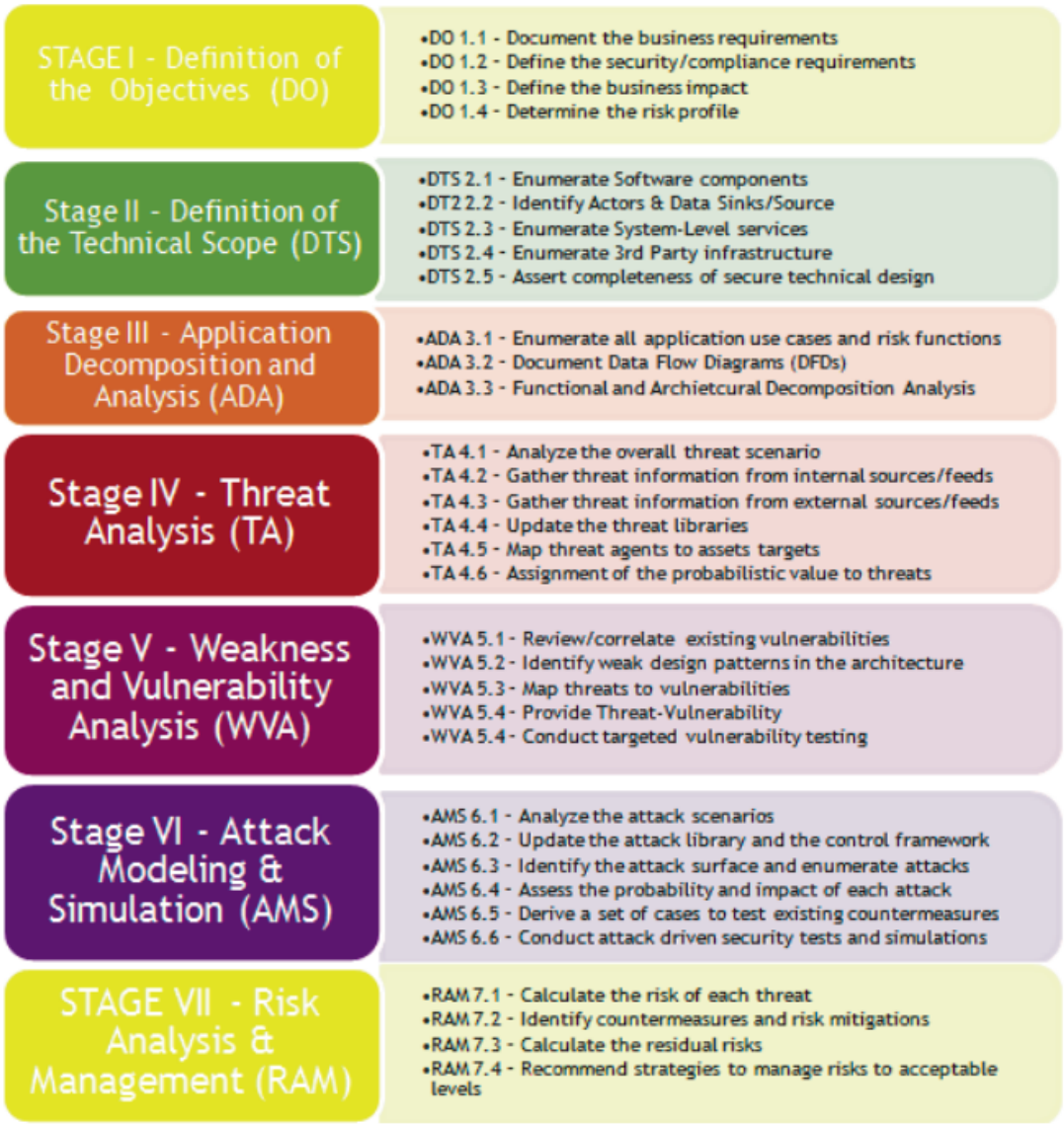

Figure 6. Stages of P.A.S.T.A (Morana, 2014)

3.8 TARA – Threat Agent Risk Assessment

Intel published their own risk assessment methodology in 2009 and it is an obvious choice when strong support for decision making is needed. Intel's TARA offers a comprehensive way to prioritize and separate important security risks from unimportant security risks and to produce situation awareness of risks for the system. TARA is used to plan and predict security risks in the early phase. The risk analysis is expected to produce accurate information about risks, and the model is planned to act as a supportive element for decision making in critical systems. The model contains also the possibility to validate the accuracy of predictions. An interesting view-

point on the model is that it is expected to adapt to continuously changing risks that is definitely benefit on risk modelling. (Rosenquist, 2009, 1)

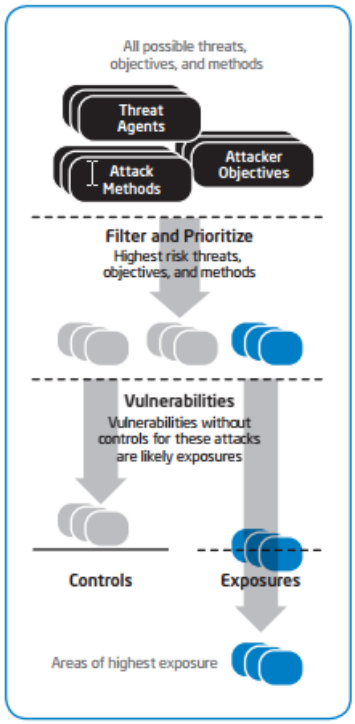

Figure 7. Identifying most important risks and threat agents (Rosenquist, 2009, 4)

There are three threat agent libraries: Threat agent library (TAL), Common exposure library (CEL) and Methods and objectives library (MOL). The threat agent library includes 8 common threat agent attributes and 22 archetypes. (Rosenquist, 2009, 4). These attributes are visibility, objectivity, skills, resources, limits, outcome and access. Each attribute contains several intentions. The archetypes are people types such as employees or spies. Common exposure library enumerates known information security vulnerabilities and exposures, and basically users should rely on publicly available CELs and add their own content. Methods and objectives library contains threat agent objectives. (Casey, 2007, 1 - 5).

As an interesting finding TARA seems to fit for health care risk evaluation to maximize security budgets based on a practical article "*Improving Healthcare Risk Assessments to Maximize Security Budgets*" (Houding, Casey, Rosenquist. 2012).

3.9 TRIKE

TRIKE is an attempt to create the methodology and the tool from the risk management perspective in a way that removes need of repetitive work. The model is released on open source community's principles with MIT license (Larcom, Saitta, 2012). TRIKE uses its own methodology that is defined by the team of security professionals (Velez & Morana 2015, 171). Basically, the model is originated from two STRIDE derivate threat models that caused frustration in practice due to repetitive tasks. Since the existing models like STRIDE did not take repetition into account the new threat model development called TRIKE was started 2003 (Larcom 2012).

The model's approach is threat modelling and system auditing with the high automation level. The model leans heavily on communication between security teams and stakeholders in way to protect the assets on reasonable level. Informing of the actions is essential and making sure that the each stakeholder understands the risks and can mitigate the risks. The system stakeholders investigate risks to the assets and mitigate the risks or accept the risks. (Saitta, Larcom, Eddington, M 2005. 1 - 3).

According to Velez and Morana, TRIKE is risk or asset based threat modelling approach. The model includes many phases like asset identification on the application environment, privileges and the communication channels that are found from other threat models. (Velez & Morana 2015, 171)

The TRIKE methodology has versions 1, 1.5 and 2. The white paper of Version 1 is on DRAFT status, version 1.5 is partially documented and version 2 is under active development. The last update regarding to the page news is from 31th July 2012 (Larcom & Saitta, 2012). According to (Larcom 2012), the TRIKE team do not have time to update the website.

Table 5. TRIKE versions and tools (Larcom 2012)

Version and the supported tool	Description
Trike v1: Squeak	• Threats are automatically generated based on the intended actions. • Threats prioritization supported. • No import or export supported. • Includes attack tree stubs (deprecated).
Trike v1.5: The spreadsheet	• Excel based spreadsheets. • Threats are automatically generated based on the intended actions. • Threats prioritization supported. • Security objectives supported. • The data collection supported without analysis. • Basic support for HAZOP analysis.
Trike v2: The Squeak (not published, the future version)	• Redesigned version • Implements version v2 • Interface expected highlight the problems and the missing information • REST interface

Microsoft's threat modelling uses STRIDE for threat modelling (SDL Threat Modelling Tool, 2015) but Trike tries to automate repetitive actions using separate threat, risk and implementation model that are based on Squeak tool or the Excel spreadsheets.

TRIKE emphasizes the high automation level, defensive perspective and the good enough security level. (Saitta, Larcom & Eddington 2005, 1 - 3).

Table 6. TRIKE phases (Saitta, Larcom & Eddington 2005. 3 - 14)

TRIKE phases	**Description**
Requirements model	Assets, actors, intended actions and rules. Actor-Asset-Action matrix.
Implementation model (DFD's)	Data flows, the actions that can be performed in real time + system state.
Threat model	Implementation model analyzation and creation of the threat model based on that. - Enumerate threats - Risk values - Attack graphs - Mitigating controls assigned to the vulnerabilities
Risk model	Based on assets, roles, actions and threat exposure.

3.10 TRIKE Usage

Usage of the TRIKE is allowed based on the MIT License (MIT) and therefore it is possible to take the model and update it according to the organizations' needs. (Saitta, Larcom & Eddington 2005, 1).

3.10.1 Requirements model

Use of the TRIKE starts with listing expected and unexpected users or user groups and assets. Simultaneously system or actually the functionality that system implements and the actions taken by human beings towards system functionality are listed. Asset stands for something that has monetary value, specific and tangible

pieces of data like system itself or the configuration file but not artefacts like passwords (Saitta, Larcom & Eddington 2005, 3-5). An asset can be a physical object if it is featured in the business rules of the system (Saitta, Larcom & Eddington 2005, 16). Human being is a self-explanatory term, however, in TRIKE it means an actor and in some rare cases other than human beings may take or cause actions and are therefore taken into account. Intended actions are something that users take towards the system. Actors, assets, intended actions and the rules are applied to the "Actor-Asset-Action matrix" to specify users, functionality and taken actions. Additionally, actions are defined by positive or negative rules that define circumstances surrounding the action. (Saitta, Larcom & Eddington 2005, 3-5).

The action control matrix or Actor-Asset-Action matrix uses database language and the TRIKE's V1 tool Squak displays matrix that uses the abbreviation CRUD (Larcom 2012). Basically it is a visual representation of requirements that eventually derivate threats as a result in automatic way. The abbreviation CRUD comes from the database vocabulary and stands for create, read, update and delete. The first square on left top corner is "C", the second square is "R", the third square on second row is "U" and the last square is "D". Each square has color based values "Always", "Sometimes", "Never" and "Unknown". The asset names are listed on the top of the columns and the actors are listed on the left side of the rows. (Saitta, Larcom, & Eddington 2005, 3-5).

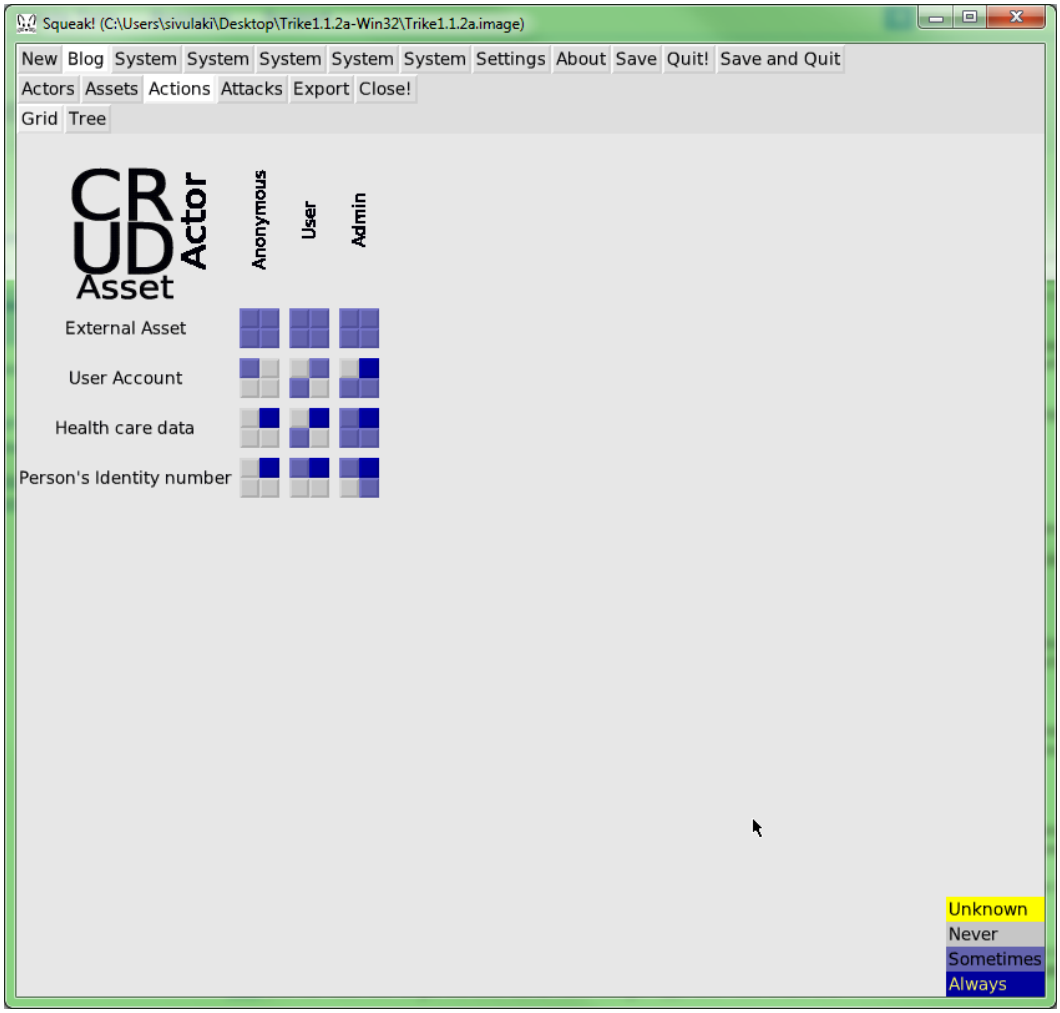

Figure 8. Squeak! The standalone tool for TRIKE V1 (Larcom & Saitta 2012, Tools).

3.10.2 Implementation model

The implementation model forms data flow diagrams of hardware and software components. The intended actions, supporting operations and the state machine are keywords for this model. Before implementation model the requirements model is finished and the intended functionality is fully understood.

The model creation starts with searching the actions that are against or do not apply into intended actions. Actions are reflected to the system states in the system with the data flow diagram. (Saitta, Larcom & Eddington 2005, 3-5).

Table 7. TRIKE implementation steps

Implementation steps	Description
List state style requirements in the rules for intended actions. Include state tag within identifier and type of the object (asset in the system, an actor (human being), specific role of the actor).	Example: Message in the inbox could be asset “Email” or “Email account owner logged in the mail system”. In this case tag could be “user”
Create the data flow diagrams (DFD)	Standard diagram type. Includes processes, data storing, external counterparts (interactors) and data flow between everything.
Create use flows (Experimental feature on Trike V1)	Use flow stands for combining the state requirements to the data flow diagrams as a use flow diagrams. Basically all actions in the system are mapped to the DFD’s.

3.10.3 Threat Model

The threat model includes threat generation, attacks, attack Trees (deprecated after first TRIKE V1) and the attack graph, weaknesses, vulnerabilities, mitigations and attack libraries. The full threat model requires the requirements and the implementation finished. The first phase is the attack graph creation and after that examination of the weaknesses in the system. The attack graph generation was intended to be automatic. The last part investigates vulnerabilities and figures out mitigations for the vulnerabilities.

Trike V1 supports two separate threats and they are elevation of privilege and denial of service. The elevation of privilege has three different subcategories: the first takes place when the actor is able to perform any action towards system that should not be possible for the actor. The second occurs if the actor performs action that is not allowed. The third happens if actor uses the system to perform the action on some

other system's asset. In TRIKE any spoofing or any tampering are considered as elevation of privilege threats. Denial of service stands for any prevention from any intended action that is described with rules as system's actions. Threat generation has only few rules. Each intended action will have one denial of service threat. The elevation of privilege threats is generated with inversion of the intended actions. These particular negative actions are selected to generate disallowed actions that are not following the rules. The last one is the social responsibility threat that stands for using the system to take the action against other system. (Saitta, Larcom & Eddington 2005, 8-9).

The attacks can be based on threat, implementation or technology specific steps. In TRIKE V1 attacks were organized in attack trees; however, the method is deprecated in later versions. The attacks are replenished with weaknesses, vulnerabilities, mitigations and attack libraries.

3.10.4 Risk model

The TRIKE's risk model is on experimental level despite of fact that is core feature of the TRIKE methodology. Basically, the risk model helps targeting resources to the right areas with help of the attack graph. (Saitta, Larcom & Eddington 2005, 8-9).

3.11 Attack libraries

The attack library is a collection of attacks for some specific purpose, and its contents depend on users, detail versus abstraction and scope. The structure can be lightweight or highly organized. The library scope defines the area where a library is applicable. At the time of writing the thesis the number of publicly available attack libraries is small because the development of the library consumes a vast amount of time and resources. The important difference between an attack library and a checklist is the threat modeling. The attack library usage always requires practical threat modeling, however, the checklist does not. (Shostack 2014, 101-102).

3.11.1 CAPEC

CAPEC is a registered trademark and an attack library within classification taxonomy of known attacks maintained by MITRE Corporation from USA. Interesting viewpoints are that the CAPEC is sponsored by U.S. Department of Homeland Security and the library is under active maintenance (Mitre, 2014). According to (Shostack 2014, 106), the CAPEC can give more results regarding to attacks than STRIDE's security properties but the CAPEC techniques are far more complex. In Velez & Morana's views (2015, 462-463) the library is one of the best and truly comprehensive attack library available. Mitre's records shows that the library contains currently over 460 attack patterns that are available within different methods like Google based search or CAPEC ID search.

The CAPEC offers a possibility to get the CAPEC library in XML format. The XML library can be used for various purposes but the parsing of the must be done locally. Basically it offers possibility to combine the CAPEC data to the tailored systems and methods and therefore it can save great deal of time because the attack library is ready for usage. However; aggregating of the CAPEC XML data within other typical attack libraries like OWASP, WASC or PTES is possible but it consumes definitely a lot of time. (Velez & Morana, 2015, 462).

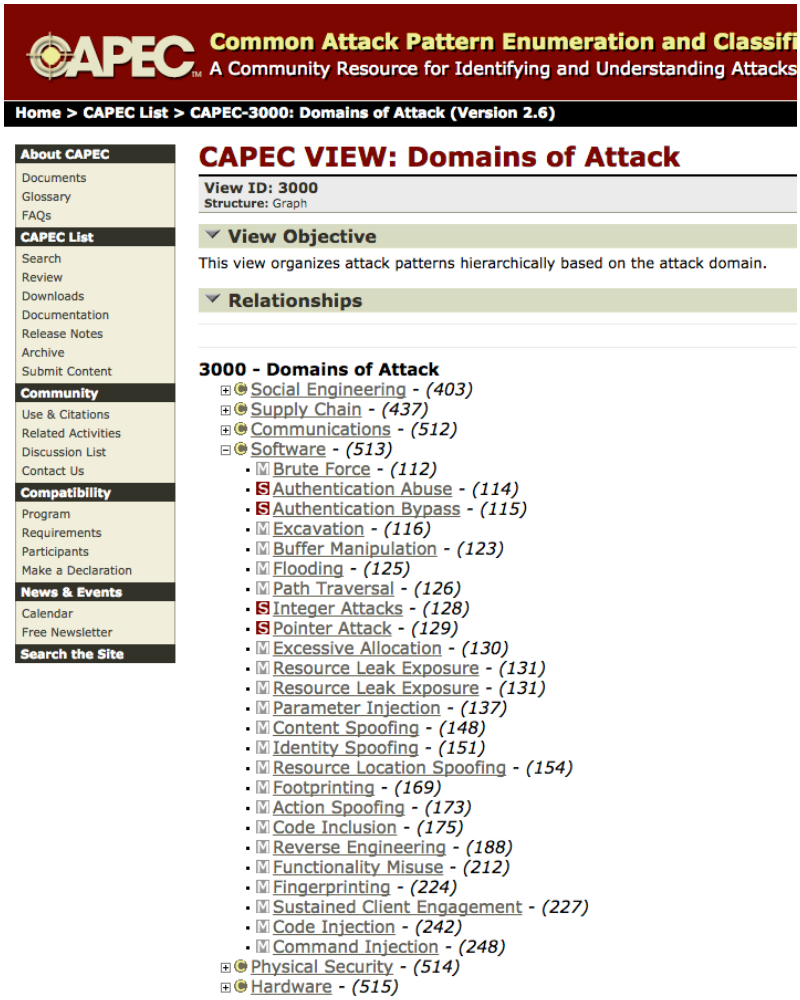

Figure 9. CAPEC VIEW: Domains of Attack (Mitre 2015).

3.11.2 OWASP

OWASP is an abbreviation for The Open Web Application Security Project. The project aims at continuously improving software security worldwide, especially web applications; however, it is possible to apply the project methodology is possible to any software project since the project consists of dozens of separate security projects. One example is the Cheat Sheet Series project that contains numerous cheat sheets for attacking and prevention. The OWASP project's most commonly known security related guidance list is OWASP Top-10 that contains 10 most commonly exploited security flaws in the web applications. In layman's terms the Top-10 project clarifies and makes security issues visible for individuals and organizations. The OWASP project focus is worldwide, and eventually security based decision making is easier based on different projects data like OWASP TOP-10 threats list, cheat sheet series or the security testing guide. (Shostack 2014, 108; OWASP 2015, Main page).

The drawback of OWASP project is that OWASP cannot work as the only security source since the project data does not fulfil scientific requirements. The project is the result of thousands of volunteer users' commits. The project does not have specific requirements for the participants, and thus the project data cannot be considered to be scientific. (OWASP 2015, Main page).

According to Shostack, the OWASP TOP-10 can replenish STRIDE threat model usage (Shostack 2014, 108). Additionally, OWASP recommends Microsoft's threat modelling process for the web applications and informs that everyone in the software development team can easily apply the Microsoft's threat modelling process. (OWASP 2015, Risk Threat modelling).

3.11.3 WASC Threat Classification

The WASC Threat Classification is the attack library by Web Application Security Consortium. The latest version is v2.0 and its release year was 2010. The library contains threats in grid view or tree view. The vulnerabilities are categorized under design, implementation or deployment. (WASC, 2010).

4 Theory research of own threat model

This part of the document describes the collected theory and background for own model before implementation phase. Theory is a basis for deeper ideation and eventually implementation of own threat model. Several different approaches are handled through behavior, crimes, attacks and threats are described.

4.1 Behaviour analysis

According to the author, nearly all of existing risk and threat models concentrate on software specific threats, however, it always needs to be remembered that people use the software, systems and computers. Jackson (2012, 99) describes behaviour

analysis as follows: "The key to behaviour analysis is to simply identify antecedents, behaviours, and consequences in a non-biased manner, although it can be difficult". Therefore, the most important model to recognize would be behaviour based analysis model and according to the author, it should be applied to the agile development models. In this thesis a deep delve into the psychology area is not possible but some noticed basic tenets are discussed.

4.2 Operant Conditioning

Operant conditioning can be considered as part of operant psychology. As a theory it consists of reinforcement and punishment and both can be positive or negative. Reinforcement simply means increasing the rate of response and punishment means decreasing the rate of response. In this model reinforcements and punishments must follow behaviour; however, their purpose is to cause different results. (McSweeney & Murphy 2006, 167-169).

Positive reinforcement can be considered as a reward and negative reinforcement as an avoidance or escape. Both reinforcements do increase the probability the frequency of the preceding behaviour. For instance, if a user commits to do something towards better security s/he may get bonus or positive attention and therefore gain positive reinforcement. Regarding to negative reinforcement user may want to wear security badge to avoid a warning. (McSweeney & Murphy 2006, 171).

Positive punishment decreases the probability of the preceding behaviour. For instance, if a user decides to install illegal software to his workstation, s/he gets caught due to workstation software audit and receives a warning of acting against company policy. In consequence of that it is less likely that a user installs the illegal software again. Therefore security policies and their enforcement in this case work as positive punishment. (McSweeney & Murphy 2006, 172).

Negative punishment decreases the probability of the behaviour with removing something due to the user's actions. For instance, a user might want to login with another user's credentials with trying to guess his / her password. After few attempts

account or even workstation is locked for 30 minutes and in order to continue right away the user must call the service desk or ask the administrator to reopen the locked account. (McSweeney & Murphy 2006, 172).

4.3 Applied Behaviour Analysis (ABA)

Applied behaviour analysis is a methodology originating from B.F. Skinner's behavioural principles derived from basic laboratory research, and the first announcement of the ABA methodology was made 1968 in the Journal of Applied Behaviour Analysis (JABA). (Carr, 2000, 295). However, there are some remarks that ABA has been around much longer. (Kearney 2007, 19). Understanding science models of psychology might reveal ways to prevent risks and threats much before they are realized. Models can be used beforehand to prevent something such as crimes or afterwards to figure out why the crime happened. Thus, in crime investigation behaviour analysis has been a part of crime investigation since 1932; however, science has improved it all the time. (Douglas, Burgess, Burgess & Ressler 2006, 4). Despite of science achievements, the basic applied behaviour analysis is still applicable and fully valid for relatively new areas like cyber security. (Jackson 2012, 4).

"*ABA is an approach to changing socially useful behaviours that employs scientifically established principles of learning to bring about these changes*." The model is often used to achieve positive changes and one typical instance is pedagogics. (Kearney 2007, 19). Applying ABA with malicious behaviour can help to recognize threats before they occur. Everyone knows that each person reacts in his/her own personal way in different situations and events. Any key event or situation can affect person's the self-image and how persons behave in the future. (Jackson 2012, 3). Eventually ABA is a part of operant psychology with the intention to help individuals and groups, no matter whether ABA can even help in situations when the threat is caused unintentionally. (Jackson 2012, 117).

4.3.1 Antecedent

Antecedents are “things that happen or are already in place before the target behaviour occurs”. (Kearney 2007, 31). According to Kearney, many of antecedents are basically neutral or not important for target behaviour; however, certain antecedents may reinforce or diminish behaviour.

Cambridge Advanced Learner’s Dictionary describes antecedent as follows: “Someone or something existing or happening before, especially as the cause or origin of something existing or happening later.” (Cambridge 2008, 53). According to Jackson, antecedents consist of preceding events and situations. (Jackson 2012, 3).

Based on previous quotations, antecedents can be considered as a result of certain meaningful or important things in past that may reinforce or diminish certain type of behaviour.

4.3.2 Behaviour

Behaviour definition depends on context but basically it is simply one or a series of acts or conductions towards others. (Kearney 2007, 23). Another notable factor is the frequency of the behaviour. (Kearney 2007, 25). For instance, in a company network a single port scan or minor security incident might not be dangerous; however, series of them require further actions to maintain security.

In Jackson’s view (2012, 165), specific antecedents with followed consequences change behaviour with reinforcing behaviour or decreasing behaviour based on following consequences. Hence, the same or similar antecedents with favourable consequences reinforce behaviour and unfavourable consequences decrease the behaviour. (Jackson 2012, 165). Behaviour is often a result of three combined factors: Heredity or genetic endowment, physiological changes and learning. (Kearney 2007, 25). According to Jackson, behaviour has always certain rules: ”The world would be chaos if all humans responded spontaneously without considering such antecedents as culture, laws, rules, convention, and patterns of past behaviour”. (Jackson 2012, 169).

In the crime investigation related studies prove that any behaviour reflects influence of internal and external factors. These factors stand for psychological motivations and social stresses. As cybercrimes are another crime class, finding the motivations and stresses should help to recognize antecedents and behaviour that might lead to certain consequences. (Douglas, A.W. Burgess, A.G. Burgess & Ressler. 2006, 383-5).

4.3.3 Consequences

Consequences are simply matters that take place after a specific behaviour occurs. Basically, consequences that follow the behaviour regularly enough change the frequency of the behaviour. (Kearney 2007, 36). Basically, the prediction of certain behaviour is associated with antecedents and consequences and not on studying the behaviour. (Jackson 2012, 258).

4.3.4 Antecedents, Behaviour and consequences behaviour chain

According to Jackson (Jackson 2012, 4), antecedents, behaviour and consequences form a chain applicable for predicting malicious behaviour by an individual or group. The model is based on behavioural psychology and more specifically on applied behaviour analysis (ABA). There is even patented automated behavioural assessment (AuBA) technology available; however, automated area is not covered in this thesis. As the model is a general viewpoint to any individual or group behavioural analyzation, it helps to prevent attacks based on collected data. The model is rooted in the past in a special program at Lincoln State School in Lincoln where a researcher (Gary M. Jackson) tried to resolve the developmentally disabled persons' behaviour problems and eventually found out the cure for their behaviour problems. (Jackson 2012, 472).

According to Jackson, news articles can reveal examples from the past that help to reveal real events and situations, actions and consequences. (Jackson 2012, 175-176). Hence, cybercrimes or attacks can be prevented with applying the usage of the behaviour chain. Searching for former crimes, violations, events or situations can help to find ways to break the chain before the unwanted consequences occur. For

instance the health care news archives may reveal the useful data that can be used to prevent the similar crimes in future. For instance, snooping of health data (access violation) is a typical crime in the field of health care and it is typically carried out by health care professionals. Antecedent might be screaming news of famous person on yellow press, and behaviour is snooping of the data from the hospital database. Naturally, the consequence of snooping could be a warning for employee or termination of the employment whenever a person is caught due to inspection or any other reason. Finding the key environmental influences that cause a definitive behaviour is the key to finding out a way to break the behaviour chain.

For instance, successful snooping attempt might reinforce the individuals to snoop data again; however, getting caught for snooping works in the opposite way. Based on the previous overly simplified example the idea is to break the behaviour chain at some point. In this case changing the consequences breaks the chain and thus makes access violation to data less favourable. As an example, detection of snooping is possible with logging all searches that an employee does with the employee id and eventually officials doing inspections searching for employees' logins. In this case, the threat to get caught due to inspections of the search logs acts as a consequence and when an employee is aware of the threat committing the crime is not favourable. Breaking the chain can happen as well in the antecedent phase and the employer can carry out security checks and specific interviews for employee to make sure the employee's background is clean before hiring him/her to the job. Despite the yellow press news or life happenings, the honest person will not likely commit snooping on another person's health data on the database.

Table 8. ABC Chain elements (adapted from Jackson 2012, 3)

ABC chain element	**Description**
Antecedent	Combination of events and situations from past that directs person's interests, choices and even moral perception.
Behaviour	Actions made by individual.

Consequence	Maintain, increase or decrease similar behaviour in future.

4.4 Motivations

Motivation behind a crime might be disgruntlement due to a certain event or situation. If a long awaited promotion is passed over to someone else or a sudden divorce is due to cheating, this might cause different kinds of motivations. The risk for cyber bullying, unauthorized data violation or even stalking is more likely to occur after certain motivations. (Jackson 2012, 212).

According to Jackson (2012, 47), "Motivation of a group is very important because it tends to solidify environmental antecedents for terrorist attacks". Terrorist group motivation is a good example of the deep motivation towards selected target. Therefore, sometimes it is more important to understand the groups' motivations instead of single person's motivation.

4.5 Modus Operandi

Perpetration of a cybercrime always requires actions, and modelling of the actions is needed to understand how the crime is committed, how to prevent its detection and how to hide the tracks. Because cybercrimes can be modelled in the same way as any other crimes, there is a Latin term intended to describe perpetrator habits in specific cases. Modus Operandi stands for "Method of operation" and for anything repeatable that describes someone's way of working and especially when speaking about criminal investigations. The criminal might have some simple modus operandi in the beginning or a highly evolved version depending on the experience and resources. (Douglas, A.W. Burgess, A.G. Burgess & Ressler. 2006, 20).

Most people want to evolve their way of working and eventually learn from mistakes. The same applies to cyber criminals since they make mistakes like any other people and they try to get better with time. For instance, a script kiddie might carry

out a denial of service attack from his / her own desktop to some hobbyist forums to disrupt them. Disruption stands for slowing down a forum user's access to the forums. However, the attacker does not understand right away that a forum's webserver has logs that store users' IP addresses and the risk of getting caught is higher than expected. Next day a discussion with other hacker friends reveals that there are ways to scramble where the attack is originated and next week he/she will attack with a distributed denial of service with distributed DNS DDoS attacks. This allows him/her more likelihood to commit the crime without getting caught by the police. DNS DDoS method allowed better attack power and thus the modus operandi evolved to the next level. (Douglas, A.W. Burgess, A.G. Burgess & Ressler. 2006, 20).

4.6 Applying behaviour analysis for cyber crimes

Investigation of cybercrimes has been a part of crime investigation since mid-1990s. (Douglas, A.W. Burgess, A.G. Burgess & Ressler. 2006, 407). Eventually applying the behaviour analysis to lone cyber attackers, groups and even to country level threats may help to prevent attacks or crimes before they do happen. It is paramount to break the behavioural chain to prevent a possible attack or crime beforehand. On the software side this stands for thinking about the possible perpetrators and their typical behaviour chains (Jackson 2012, 175-176). For instance, a cyber-attacker might have experiences from past that attacking some website was fun and very interesting.

A typical example to inspire an attacker could be news where young attackers managed to bring down Nordea's website (Sky news, 2014). Due to this inspiration, the attacker decides to plan malicious attacks and finds out other hackers from the Internet. These hackers start to collaborate and learn hacking by doing. Later they may decide to attack some specific site. The group does reconnaissance actions and maybe scans the website. After reconnaissance is done the attack can start. In the beginning the attacker might feel fear of being detected; however, s/he may notice soon that it is easy to carry out denial of service attacks and not get caught. Eventually s/he advances and manages to exploit some existing security flaw to get into someone's personal details (Jackson 2012, 175-176); Walker 2014, 49-50).

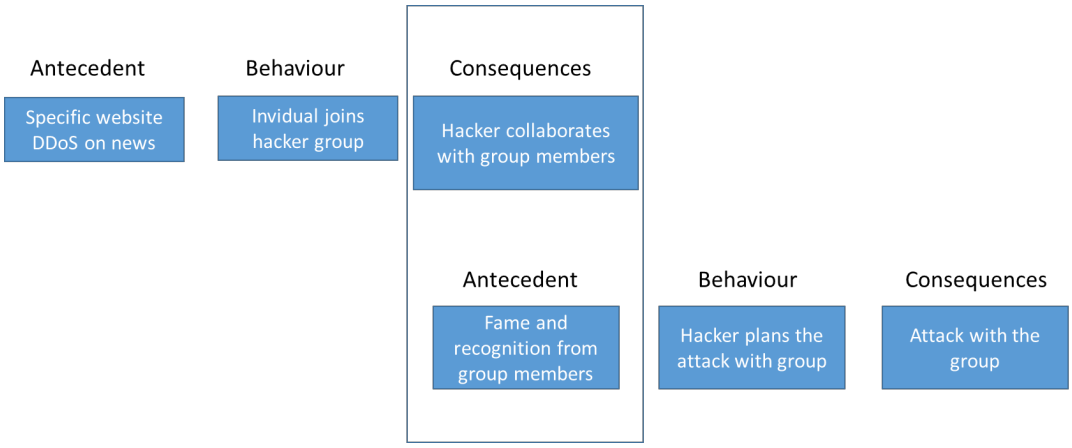

Figure 7. Behaviour, consequences and antecedent chaining (adapted from Jackson 2012, 8)

Ultimately, the prevention of attack requires cutting the behaviour chain in some phase. For software it would stand, for instance, for increasing the danger of getting caught with adding Security Information and Event Management system, a separate log server, warnings to website for criminal liability for cyber attacking and continuous follow-up of the malicious activities. For single cyber attackers increasing the risk to get caught (consequences) might be enough to prevent the access violation or even the crime (Jackson 2012, 8).

4.7 Attacks

Sun Tzu wrote “All warfare is based on deception” (Giles, 1910, 3). These days Russia performs very fuzzy politics in Ukraine, European area and borders of USA. Basically, all this behaviour includes deception, propaganda and changing mental views. Everything is done only to get the best advantage for a malicious country (Yle, 2015). Deception itself applies to cybercrimes as well as espionage or cyber theft. Deception is exceptionally important for the attacker because when the attack is easily noticed then the risk to be caught rises. Therefore, the attacker may be very deceptive and use for instance a combination of exploits and social engineering to achieve targets. Basically, a crafted and quiet attack enables an attacker to achieve the target without anyone knowing about it. (Jackson 2012, 18).

4.8 Attack types

According to Walker (2014, 19), there are four different attack types that a hacker can use and these apply only to computer software and networks. The following table illustrates four different attack categories and descriptions for them. (Walker 2014, 19).

Table 9. Attack types According to EC-council (Walker 2014, 19).

Attack type	Description
Operating system attacks	Attacks regarding operating systems without latest security patches, default security settings, weak passwords or no password at all, no firewall, forgotten user accounts, no encryption on the hard disk and so on.
Application-level attacks	All logic and code of the software. Maybe the easiest place to find vulnerabilities.
Shrink-wrap code attacks	Exploitation of available automation and installation tools' code and scripts.
Misconfiguration attacks	Configuration exploitation with intentional or unintentional misconfiguration.

4.9 Hacking

When discussing hacking there are different kinds of hackers and there is a number of different categories for them. In networking field hacking is about efforts to find ways to break, prevent access or steal something. One publicly known categorization is using the terms white hat, black hat and grey hat hackers. White hats are doing well good in an ethical way and they always have the permission from the customer for hacking their systems. Black hats have also another name, crackers and their intention is to steal, destroy or deny access to systems or resources. They definitely do not have a permission for the hacking. Grey hats are something between white hats

and black hats. Their definition is more or less neutral; however, often they turn to the black side due to the lack of the permission. Some grey hats are just curious and hack because they feel that it is good for everyone; still they are doing it without permission. (Walker 2014, 18-19).

4.10 Hacking phases

Recognizing security risks and threats from networks needs basic understanding of how the attacker typically may work. Certified Ethical Hacker certification contains ethical EC council's five-step hacking phases that is the basic model of a full scale hacking case. It is important to notice that an attacker might use only some steps and go straight forward to gaining access phase after getting a perfect opportunity. This model should help to recognize risks because thinking like a hacker may reveal completely new viewpoints.

Table 10. Hacking phases According to EC-council (adapted from Walker, 2014, 23).

Hacking phase	**Description**
Reconnaissance	Most important phase to recognize useful weak points. Collect useful information of the target with different methods like phishing, search engines, social engineering or dumpster diving. Can be stealthy or not depending on the case.
Scanning and Enumeration	Deeper technical reconnaissance based on reconnaissance phase. Enumeration of the target hosts.
Gaining Access	Attacking the target(s) based on scanning and analysis of enumeration results.
Maintaining Access	Create backdoors so access is permanent.
Covering Tracks	Hide all the tracks possible so that nobody notices the successful attack.

For instance, an attacker may want to find out the easiest way to hospital's internal network to get data from patient registry. The most favourable access would be a remote connection – however, that is easier to say than do. In reconnaissance phase s/he investigates everything about the hospital from the Internet and collects data fragments. In a later phase a visit to the hospital may follow with some excuse a visit to see a doctor. The idea of the visit is to find out how to connect a laptop to their network internally – after trying some doors the attacker finds some network sockets that seem to work. Additionally, s/he searches for interesting papers from trashes with dumpster diving. On some other day he might call the hospital and ask deceptive questions from the hospital's employees using social engineering. In scanning and enumeration phase a laptop is simply connected to hospital's network because few suitable network sockets were found in reconnaissance phase and s/he already knows how to access them. Gaining access is the next step and eventually the access is achieved from hospital's network because it is much easier to bypass the hospital's firewall that way. Maintaining access phase succeeds with the installation of backdoors to old Windows XP computers, and the hospital's network can be accessed from remote connection. After the attack is completed the tracks are removed with the deletion of the backdoors.

4.11 Three major forms of attack

Classification of the attack forms might be an endless task but approaching the problem with three major forms of attacks helps to classify threats on a general level. Jackson approaches the growing damage and threat with these three forms and basically they seem to be applicable to any attack type classification from internal / external point of view. These three mentioned forms are external threat, trusted insider and insider without intent. (Jackson 2012, 117).

4.11.1 External threat

“The external threat from a hacker or group of hackers intent on causing damage or stealing restricted information”. (Jackson 2012, 117).

The external threat is probably the most commonly noticed threat on the news and it is often more clear how to prevent the external threat. For example, an attacker could bypass a weak login procedure and try to steal personnel data.

4.11.2 Trusted insider

“The trusted insider who goes bad as a result of disgruntlement, or the promise of compensation for stealing inside secrets (insider with intent)”. (Jackson 2012, 117).

Often insider employers in companies and organizations forget that the most dangerous threat can be an insider. That is worth of consideration because an employee may have access to classified information or proprietary information and share it forward or even sell it to someone who needs the data. For instance, someone’s spouse is very jealous about an ex-fiancée and wants to see her health data because he wants to damage her reputation with gossip talking. The internal employee may use some systems at her workplace to search data from different systems and collect some data pieces to start gossiping on Facebook. Often such cases end with an internal audit and they employee may lose her job, which is considered as cyber theft and cyber bullying.

However, it is not so simple in all cases. Sometimes an insider might be clever and seek a partner to get assistance in delivery and distribution of restricted material. This can be the case especially when the amount of data is very big or the data is really sensitive. In the wildest scenario the insider is recruited by foreign intelligence service; however, that is not the case with ex-fiancées. (Jackson 2012, 83).

The trusted insider always uses insider methods of operations (Jackson 2012, 285). For instance, the thief could first gain access, locate the valuable asset in the net-

work, steal the asset from the network, store the stolen data and pass it forward. A good example of the suspected thief was a recent Ashley Madison database leak by female who worked for Avid Life Media (McAfee 2015).

4.11.3 Insider without intent

"The insider working in collusion with or being duped by an outsider to facilitate damage and loss (insider without intent)" (Jackson 2012, 117).

Sometimes an insider does not know to be part of the crime and it also happens in health care area. Understanding the dark side of people's intentions may reveal very clever attacks. A typical example could be a social engineering attack where someone gets a new friend from the organization and after a nice beginning the attacker tries to get classified or sensitive data from the insider without intent. A well-crafted social engineering attack might take place very slowly and the insider without intent does not even know that s/he has been a part of the crime. Sometimes social engineering may occur with dating someone from the organization. When someone falls in love they might be easy targets for social engineering attacks; however, then the price must be very precious. (Jackson 2012, 117).

4.12 Generation of new attacks

Generation of new attacks takes place in three approaches: modifying existing attacks, creating them from scratch or getting all the organization data and building an attack based on that. From national point of security views maximization of cyber-attacks needs also factors that help to boost an attack; however, it is applicable in a much wider scale. The first factor is that attacks are a surprise and therefore completely unexpected. The second and even more important factor is to have a new attack form to avoid recognizing it. The third factor is that the attack needs to be harmful. The fourth and last factor is that the attack should be well camouflaged and thus deceptive. A deceptive attack form is particularly important on the national level (Jackson 2012, 128 and 147).

The generation of new attacks is not easy task but the skilled attackers with create attacks from scratch and avoid the detection. (Jackson 2012, 128).

Existing attacks are good starting point for modifying a new attack. Attacker just selects existing attack, modifies it and use it. Idea is to modify the attack so that the signature detection does not notice it. (Jackson 2012, 128)

4.13 Attack detection

As attackers tend to be deceptive there must be ways to detect the attacks. One way to detect attacks is to use security audits. According to Vatanen (2014, 89), an attacker might eavesdrop systems and conceal their existence completely without no-one knowing. On the other hand, Finnish privacy law is not clear enough and therefore it prevents data collection thoroughly so different tactics are required to prevent cyber theft and espionage.

4.14 APT – Advanced Persistent Threat

The most dangerous cyber threat is an advanced persistent threat. Basically it can be formed by one or more people and it is often extremely well crafted for some specific target or entity. Highly advanced viruses have been on the internet for many years and often they originate from national level. Examples of countries typical to use APT attacks are China, Israel, Russia and USA; however, the list is not limited to these. APT can be very deceptive and even trained security personnel will not notice them easily. Basically, searching for them is like finding a needle in the proverbial haystack if there is no clue of the most valuable assets (Advanced Persistent Threat 2013, Cole).

4.15 Cyber crimes

Crimes are carried out to achieve benefits with something desirable or cause something bad to victims (Jackson 2012, 272). Few decades ago personal data theft was possible for instance by copying or taking photos of documents and this took place

on numerous espionage cases. Based on global threat analysis (Jackson 2012, 141-142), one of the highest probabilities for future crimes is cyber theft and basically the reason behind of that is that the adversaries use previously unknown patterns to avoid signature detection (see chapter 4.16).

Jackson describes cyber attack or cyber theft by following clause: “Cyber attack / cyber theft is strong alternative that affords anonymity” (Jackson 2012, 78). Basically it stands for that attacker is often unknown especially when using the Internet environment, however; it applies also to internal threats since insider can commit cyber attack with all the available information. Cyber theft itself is not a new threat; however, as health care services are more and more on the Internet the probability of theft rises.

Technical network security is reactive and thus not safe enough to protect the data. Nevertheless, it does not stand for forgetting reactive aspects but instead of that moving the focus on preventive aspects.

Targeting health care area is expected from curious, jealous, hostile or even terrorist types of people. In the worst case loss of lives may follow if known patterns are exploited. Finding flaws takes place with recognizing and understanding the patterns of health care processes. In the field of software processes it is essential to understand these patterns and continuously follow-up and inspect processes. Ideally the target is to recognize possible flaws in software and more importantly, in the overall processes regarding the software that implements health care processes. The probability for cyber theft is now frequent – even a daily matter - and an adversary can be anyone; however, in most dangerous context they can be state-sponsored entities, foreign intelligences and intelligence gathering. (Jackson 2012, 142).

4.16 Signature detection

Signature detection is the recognition of known patterns on a specific context. It is often seen as a part of cyber context; however, in reality it can be used nearly in any

other context (Jackson 2012, 141). In cyber context for instance network traffic is a series of bytes, and bytes are combined to packets. Typically known remote attacks contain a known signature that is built of the known series of bytes, and this signature is easy to form of the byte series; however, modifying the attack causes the situation when the signature is different and therefore not detected nor necessarily prevented anymore. Signature detection can be seen as a pattern model that supports former known attacks or security violations done in past. Signatures are also prone to false negatives and especially in systems where the signatures are overtuned. (Jackson 2012, 80).

A horrifying example of exploiting the known signature detection was Germanwings plane crash where Andreas Lubiz locked the cockpit door when the captain left the cockpit and after that flew the plane to the mountain and killed 150 people in the French Alps. (The Guardian, 2015). The captain could not get into the cockpit because of a flaw in the known attack signature. 9/11 attack caused strict procedure changes to cockpit door security that have existed since 1980s; however, it could not take into account that cockpit must always have two pilots. (Tribute 2015). It is not an example of a cyber attack; however, it presents a perfect example on how to exploit known patterns and cause disaster with exploiting known patterns. Signature detection is always based on data from the past and it covers basically only known attacks, and therefore it is always one step behind the attackers. (Jackson 2012, 80).

Human expertise is essential to craft proper signatures and rules for anomalies. Basically, the problem is that machines can apply different algorithms, recognize signature and report found anomalies; nevertheless, human expertise is still required in forensics to recognize evidences. (Jackson 2012, 80).

4.17 Anomaly detection

Anomaly detection means identifying that something acts in an unusual way or something is not considered normal when comparing to known data of the target. For instance, it is no anomaly if somebody wears a dinner jacket when s/he is going

to play football. (Jackson 2012, 266). Anomaly detection uses statistical science approach to find out differences from typical network traffic averages (statistical norms). The network traffic can be tracked continuously, and statistical data of typical network traffic can be collected around the clock. For instance, it could be normal that employees use FTP during work time in company's internal network and therefore its normal. However, it is an anomaly if someone starts FTP connection to China in the middle of the night from the company's network. Reflecting on Andreas Lubiz's locked cockpit door example in March 2015, the captain leaving the cockpit was an anomaly, however, it was noticed only after the plane's black box investigation by the French officials. Basically in such a situation there is nothing malicious, nevertheless, that is not the whole picture.

Anomaly detection has its benefits to notice anomalies; however, the problem is about indicating malicious activities even when there is nothing malicious going on (Jackson 2012, 269). When reflecting on the captain leaving the cockpit the anomaly exists, however, before this there was nothing dangerous based on the earlier experiences and therefore the former anomalies have been completely fine. Now the anomaly was the captain leaving the cockpit and the first officer's intention to lock the captain out of the cockpit. Making the difference between whether an anomaly is good or bad is very difficult and even harder when all details are not known. (The Guardian, 2015).

4.18 Computer crimes

United States of America is seen as one of the promised lands in the cyber security protection due to its political and national position in the world. Computer related crimes have caused a need to create a new classification for them and even new section that prosecutes on the high technology, computer crimes and intellectual property offences. The mentioned section is U.S. Attorney's Office's Computer Hacking and Intellectual Property Section. (Douglas, Burgess, Burgess & Ressler. 2006, *383)*.

Also, the crime classification guide divides computer crimes into four separate classifications: ”*the computer as the target of the crime (510), the computer user as the target (520), criminal enterprise (530), and threats via the Internet (540)*” (ibid., 4)

Table 11. Crime classification manual list of computer crimes in USA (adapted from (Douglas, A.W. Burgess, A.G. Burgess & Ressler. 2006, 383-403)

Crime type	**Description**
510: Computers as the target	The victim is the computer itself (not hardware) and especially the data or the software stored on the computer. Target can be the user, the trade secrets, the intellectual property, the data or the software.
511: Malignant software	Any harmful or malicious software like malware, Trojan horses, viruses, logic bombs and worms.
512: Computer data as the target	Unauthorized tampering like changing data, replacing data or creating new data on the computer. For example, money transactions or stock information are typical targets of the crimes under category 512. The category includes the software piracy and the theft of intellectual property.
512: Denial of service	The target is attacked with denial of service by the offender.
520: The computer user as the target	Identity theft or fraud due to financial gain. Stalking (harassing through Internet).
521: Identity theft	The identity data stolen from user’s computer using the internet. The identity used for the other crimes. For example:

	Social security number, credit card, PIN codes. Includes phishing. Does not include “dumpster diving” or theft of wallet or other item with important data.
522: Invasion of privacy	Unsolicited sexually explicit material or spam through Internet.
523: Cyberstalking	Cyberstalking is synonym for the stalking. However; the stalker uses the computer to follow and stalk the victim.

530: Criminal enterprise	Criminal organization crimes with a computer or Internet. For example software piracy, intellectual property on electronic media, child pornography and money laundering belongs to this category.
531: Money laundering	Money laundering tries to make illegal funds to legal.
532: Child pornography	Child pornography on computers and networks.
533: Internet fraud • 533.01: Bank fraud • 533.02: Fraudulent Internet transactions	Computer usage for fraud like credit card or bank account fraud. Also counterfeits of the ID cards or passports falls to this category. 01: The internet money transfer between bank accounts with any fraud. 02: Purchasing of the goods but the buyer never receives the goods due to fraud.
540: Threats via the Internet	Threating messages delivered through

	Internet. For instance a computer user, family or friends are threatened with bodily harm or destruction of something.

5 Implementing the research project

The following chapters contain the description of the implementation of the research project. Initial state, first approach with: ABA attack library, the second approach with STRIDE and CAPEC, Elevation of privilege card game description, the game rules and the implementation summary. The creation of the author's own model is a part of the implementation, and it is located in the next main chapter

5.1 Implementation summary

The literature analysis approach and in the end practical workshops with a self-completed questionnaire as a secondary method generated the backbone for this thesis. The research project was organized into 9 main steps (Table 12), and planning and implementation followed the research questions as much as possible. The research questions are listed below for the sake of clarity.

Research questions:
What appropriate risk and security models already exist?
Which existing risk and security model fits best for health care software development processes?
What are the weaknesses of existing models?
Does the developed health care specific risk and threat model offer better risk and security management than generic models?

The following table contains the research steps and a short description of every research step.

Table 12. Research project steps

Stage	Description
Literature research	Collected data and references from literature and electronic sources. Creation of thesis template from scratch.
Analysis and interpretation of found risk and threat models	Analysis of data, interpretation to the thesis. Preliminary evaluation and discussions with the employer.
Ideation based on analysis and interpretation	Collected ideas and continued literature research based on new data.
Development of the own threat model	Initial ABA attack library model developed. Development cancelled after first version.
Selection of 2 – 3 most suitable methods for agile development teams	Selected STRIDE and CAPEC based on literature analysis and feedback from employer.
Planning the practical workshops for development teams	Approach with gamification for STRIDE and learning by doing approach for CAPEC.
Practical workshops	5 workshops held in Finland and Sweden. One workshop cancelled in Norway and 2 workshops in Ostrava.
Respondents empirical data analysis and literature data final interpretation and connecting it to the research questions	Main analysis and recommendations based on collected data.
Conclusions and further research	Finalize thesis.

5.2 Initial state

The teams did not use specific risk or threat models, and the existing process did not take threat modelling into account. Some professionals got security area knowledge in the organization, and employees have participated in security workshops. Security

audits are part of the software security. All new software is produced with modern software development methods following contracts, regulations and laws but organization needs to take security to part of daily development processes. Modern software development methods mean agile and lean methods in this case; however; due to employer's co-operating negotiations in autumn 2014 the business scope of the thesis was internally updated since the author was moved to Tieto's internal Industrial internet start-up. Therefore, the practical part of the thesis contains a team from Healthcare and multiple teams from Industrial internet. Both units use lean and agile methods. However; since the Industrial internet started in autumn 2014 the organization maturity was in completely different state than the Healthcare organization.

5.3 First approach: ABA attack library model

The writing of the literature review and initial thesis version started in July 2014 and in a month the first set of data was built up. The idea was to evaluate the models in October - November 2014 with healthcare development teams, since Healthcare's calendar looked optimal for practical tests in the autumn; however, the author's calendar free time in autumn was filled with a surprisingly demanding cyber security exercise course as the author was part of the red team that had to build game scenarios with the white team.

The thesis was restarted in February 2015 when the first idea of an own model was invented, and in spring some sketches were created. At the same time the author started to have meetings every second week to get the latest status updates of the work. The first real model on the thesis was invented and written in March 2015 – May 2015; however, in one weekly status meeting in May it was decided to cancel that and concentrate only on the existing models. That was frustrating since the author had spent almost 4 months on the model creation. However; as a result the former literature review offered many applicable sources for further evaluation, and the second attempt started with full speed. The last updates and changes to the own model were made in November 2015 due to model walkthrough when the thesis was submitted to the university.

5.4 Second approach: STRIDE and CAPEC

The second approach started with another literature review, and since the first approach material was not anymore applicable, it was not clear at all which models would fit for healthcare purposes. The answer was found from agile models world since they concentrate a great deal on doing instead of strong documentation. A vast amount of models were abandoned since they demanded too much from the team or organization.

DREAD was interesting and one idea was to use it since it was quite straightforward to use; however, Microsoft abandoned the model in 2010 so that was a strong indication not to use the model. The model is documented in the thesis if the teams want to use it nevertheless. Brainstorming methods were easy but security context was missing and the author bases his opinion on literature that using brainstorming only does not help the security in an organized and constructive way.

Microsoft's threat modelling methodology is definitely useful but it is much more than just threat modelling. It is an actual process family that helps to build complete security during software development, due to which Microsoft SDL was ignored. STRIDE was the first model that could fit for agile software development since it can help any software development team to build good defenses with a small effort. DESIST was an interesting STRIDE variant and even though it could fit for agile development the author could not find enough material or literature regarding to that, thus, the model was ignored.

P.A.S.T.A was a strong attack simulation process and it was the most promising to improve security; however, it was way too heavy for small agile teams and due to that it was ignored. TARA was found in September 2015 and was not selected to tests since found it too late. Based on materials it could be a very good model for the purposes of health care and agile development team; however, practical tests and some material building is needed.

TRIKE was as well a promising model and its advantage was threat modelling automation. However; it was based on Excel spreadsheets, and there was no literature regarding to the model so it was decided to ignore it since it was not mature enough.

CAPEC was the only big and clear attack pattern library but OWASP TOP-10 was also a good choice. However; it was decided to go with CAPEC since Tieto's healthcare organization products are much more than just web applications. Eventually it was decided to select STRIDE and CAPEC for further evaluation. OWASP TOP-10 was the third model but it was decided to leave it out due to the project timetable.

5.4.1 Workshops ideation and planning

After STRIDE and CAPEC were selected as primary methods planning and preparation activities for workshops had to be started. The first idea was to have a PowerPoint presentation and additionally one A4 description per each STRIDE element. After that the teams were expected to carry out practical tasks; however, based on earlier experiences many people tend to get frustrated and give negative feedback for such workshops. Therefore it was decided to find out something more interesting to avoid workshop with negative results.

Based on literature analysis Elevation of privilege card game was found, Blackhat presentation was watched regarding to the EoP game and after that it was decided to use it instead of using a typical and obviously boring teaching method, Hence STRIDE workshop part decided to build on a gamification approach with elevation of privilege card game. Since CAPEC was just an attack library it was decided to present it with a short PowerPoint presentation and then give the task to workshop participants to link the applicable attack patterns to the findings. In the planning phase the author did not know what the reality would be.

The elevation of privilege card game was found from The Game Crafter web shop; however, the price was an obstacle for multiple card decks. (The Game crafter, 2015). It was decided to print cards with colour printer and laminate them at home.

Totally 3 card decks were printed and laminated in few days. Two decks are in Sweden and one deck was used in other workshops.

The first workshop was partially a failure since too short amount of time was reserved for the workshop. There were some unclear issues and it was decided to fix them by the upcoming workshop. The second workshop was held in Linköping and was a successful case. The third workshop was held in Helsinki with an architect team and it was successful as well. The fourth workshop was held remotely with Karlstad and it worked fine, nevertheless, it was decided that it is the last remote workshop since there were too many practical problems. The fifth and last workshop was held at Oulu with Tieto's Healthcare area product team. One team in Norway cancelled the workshop and two workshops were cancelled in Czech.

Based on good feedback the real card decks were ordered from The Game Crafter webshop to Tieto's Industrial Internet unit.

5.4.2 Workshop structure

The workshop consumes one working day; however, the first workshop was held in 4 hours, which was a way too short time. For other teams the workshop length was about one working day with breaks and lunch between 9:00 – 16:00.

Preliminary requirements:
An application diagram with API interface is needed to describe the interfaces on a generic level. The elevation of privilege game is probably easiest to start with a generic architecture picture with 3rd party connections. Another possibility is to use whiteboard: one team decided to use the whiteboard instead of a drawn picture as a facilitator instructed that the team should select at least one figure for the workshop.

Agenda:

1. STRIDE PowerPoint presentation and few supporting videos for the workshop. Length maximum one hour.
2. Finding threats with Elevation of Privilege card game

3. Microsoft’s threat modelling tool
4. PowerPoint: CAPEC. Length maximum 0.5 hours
5. Apply CAPEC's attack patterns on the findings that found during the Elevation of Privilege card game.
6. Submit a questionnaire and answer the questions.

Background data for the workshop:
This workshop is part of my Master's thesis for Tieto's healtcare. The name of the thesis is "Security Risk and Threat Models for Health Care Product Development Processes". Its purpose is to research existing risk and threat models and find out a way for agile teams to improve the security on software development. The assignment was given by Tieto's healthcare (Jyväskylä) by Heikki-Pekka Noronen.

University:
JAMK University of Applied Sciences
http://www.jamk.fi/en/Education/Technology-and-Transport/Information-Technology-Masters-Degree/

5.4.3 Elevation of privilege card game

The Elevation of privilege is a card game for learning the STRIDE threat model. The game is a close variant of trick taking card games but the deck does not contain traditional suits spades, hearts, diamonds or clubs. EoP uses STRIDE threat model suit (Spoofing, Tampering, Repudiation, Information disclosure and Elevation of privilege). The purpose of the game is to teach security in a fun way with STRIDE’s security elements.

Figure 10. Colour printed and laminated Elevation of privilege card deck

5.4.4 EoP rules with CAPEC mapping

This variant of the original EoP game is played clockwise and the rules are modified by the author based on practical experiences during the workshops and the received feedback on questionnaire. The optional amount of players 3 – 6 and the players need to agree who fills the findings and score to the spreadsheet. The spreadsheet should contain columns Name, Points, Card, Components, Notes on threat, CAPEC link and JIRA ticket / bug report / feature request.

Game goals:

- The goal is to start looking your software security design with STRIDE as easy as possible
- Raise discussion about the threats towards your system or its application interfaces
- Have fun and learn with gamification approach style

Points and winner:

- +2 for a threat on your card if you can address it (Use timer to set max 2 – 3 minutes time to think about the threat possibility.)

- +1 for threat if another player can address it right after first player stops or time is out.
- +1 for taking the trick (with 5 cards it is possible to gain 5 tricks since each round have always winner)
- With 5 cards the game has 5 rounds
- Winner is the player with the highest score (possible prize can be decided by team)

Rules:

1. The dealer shuffles the cards face down.
2. The dealer shares 5 cards to each player
3. Play starts with the card 3 of Tampering.
4. Player reads the card and announces the threat. Player scores if (s)he can address threat described on the card.
5. Scorekeeper fills the spreadsheet and possibly gained score down.
6. Each player in turn follows the suit if they have a card in the suit.
7. In each round the high card takes the trick or with Elevation of Privilege taking precedence over the suit lead.

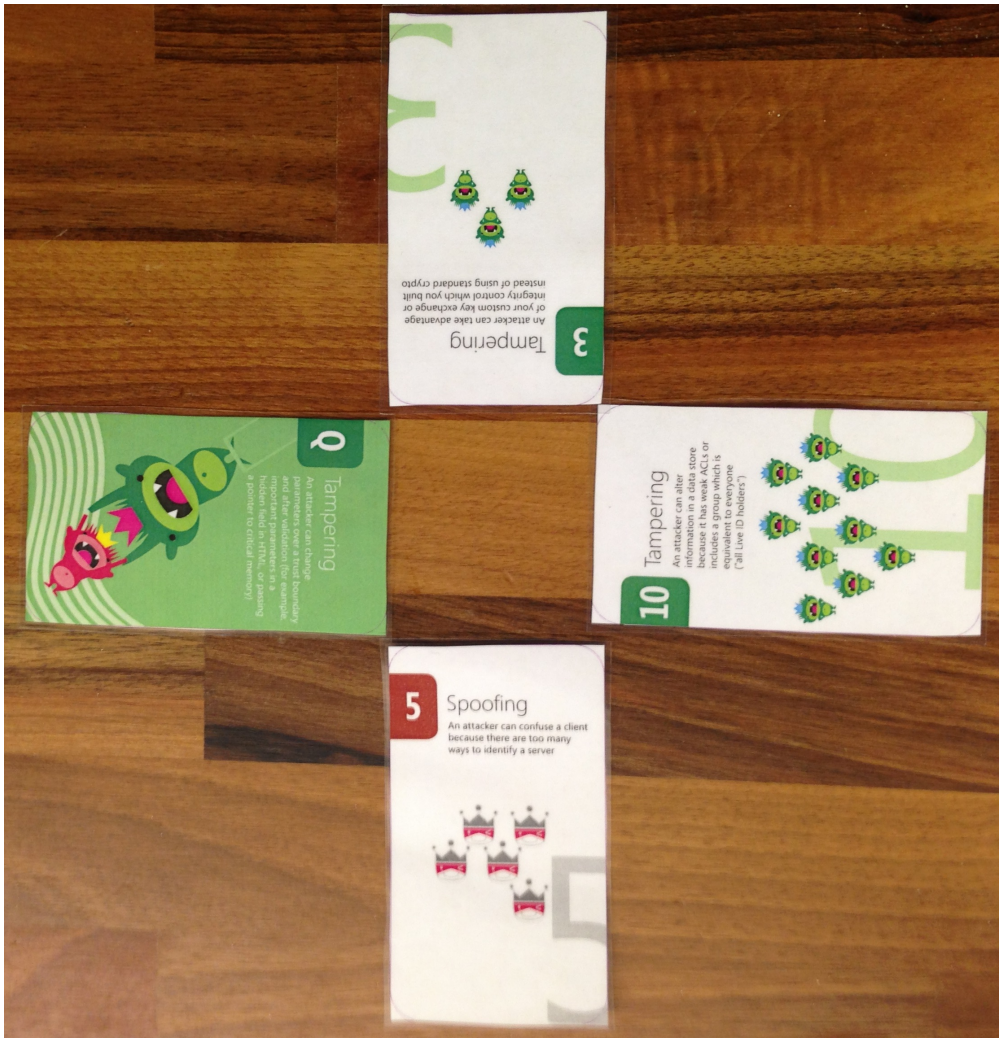

Figure 11. Queen of Tampering wins the trick

5.4.5 Creation of Questionnaire: Security risk and threat models

Originally the first version of questionnaire was written in Word; however, moving the question set to JAMK's Webropol was a surprisingly difficult task. Then it was tried to create a second version with the tool; however, the application usage was not as straightforward as expected and also the new version was tried but certain question types were missing, after which it was decided to go for Google Forms. The author managed to create the final version questionnaire in two working days after few iterations. The question set reflects on the research questions but its main points were to figure out the compatibility for agile development, how easy the model is to understand and apply, and how much it helps to recognize risks, threat, vulnerabilities, weaknesses.

5.4.6 Implementation summary

The literature analysis approach and in the end practical workshops with a self-completed questionnaire as a secondary method generated the backbone for this thesis. The research project was organized into 9 main steps (Table 12), and planning and implementation followed the research questions as much as possible. The research questions are listed below for the sake of clarity.

Research questions:

1. What appropriate risk and security models already exist?
2. Which existing risk and security model fits best for health care software development processes?
3. What are the weaknesses of existing models?
4. Does the developed health care specific risk and threat model offer better risk and security management than generic models?

The following table contains the research steps and a short description of every research step.

Table 12. Research project steps

Stage	**Description**
Literature research	Collected data and references from literature and electronic sources. Creation of thesis template from scratch.
Analysis and interpretation of found risk and threat models	Analysis of data, interpretation to the thesis. Preliminary evaluation and discussions with the employer.
Ideation based on analysis and interpretation	Collected ideas and continued literature research based on new data.
Development of the own threat model	Initial ABA attack library model developed. Development cancelled after first version.

Selection of 2 – 3 most suitable methods for agile development teams	Selected STRIDE and CAPEC based on literature analysis and feedback from employer.
Planning the practical workshops for development teams	Approach with gamification for STRIDE and learning by doing approach for CAPEC.
Practical workshops	5 workshops held in Finland and Sweden. One workshop cancelled in Norway and 2 workshops in Ostrava.
Respondents empirical data analysis and literature data final interpretation and connecting it to the research questions	Main analysis and recommendations based on collected data.
Conclusions and further research	Finalize thesis.

6 Creation of own model: Applying ABA on the agile models

Creation of the own model is based on fourth research question. However; an original idea was to modify the existing models or even combine them. Understanding threat modelling on healthcare business area seemed very specific case and in spring 2015 none of the found models did not fit perfectly on the area. Ideation started around agile models, threat and risk based literature review and searching of other theses from Theseus.

The agile methods on software development area took place after the millennium (Manifesto for Agile software development, 2001). The agile methods are nowadays commonly used models in software development houses and they have recently taken over of the waterfall style development models. For instance, few popular agile development models are Scrum, Kanban and Extreme programming (Noronen, 2014, 14 – 22).

The author's own model is attempt to plan compatible threat library to apply with the agile development methods. Users can decide the agile model they want to use. Originally got an idea for the own model by coincidence during literature review. Found a book from school's library that was about predicting malicious behaviour (*Predicting Malicious Behaviour: Tools and Techniques for Ensuring Global Security* Jackson (2012)) and it includes guidance for the patented tools to predict malicious behaviour in global security. However; idea of the own model formed first time when read the successful treatment story for the seemingly intractable problem suffered by a mentally disabled person (Jackson 2012, 476). Eventually idea formed when read Pasi Koistinen's master's thesis about security development model for agile teams (Koistinen, 2013, 93).

Based on study of ABA model and existing threat models including author's real life experience of the agile methods leads to possibility to develop a risk or threat model that should reveal most of major business specific security flaws during software development. Eventually, the model applies operating conditioning to the software development processes on high level.

6.1 Combine the software requirements, agile development and ABA

Based on the literature review there are no antecedents-behaviour-consequences chain applied with software development methods before this thesis. The Jackson's tools and techniques are definitely useful for global usage but they are too heavy and inconvenient for agile software development. According the author the agile software development needs light and easy threat model. However; numerous scrum or other agile method based security models exists like Microsoft SDLC, Cisco SDLC or OpenSAMM. In Finland Pasi Koistinen has evaluated own security model based on 2 years research work. (Koistinen, 2013, 93).

According to the author, previously listed models are considered more or less technical but they do not take peoples' behaviour into account in deep manner. The mentioned models contain a wide diversity of elements too comprehensive for a typical agile development lifecycle. Despite of agile software development large sys-

tems may have dozens of different agile development teams and in such case upper level architecture lead is required. Another important viewpoint is that technology is not the only way to stop the attacks. Instead of that, policies can answer to many threats. (Jackson 2012, 129). Therefore the risk or threat model that is based on ABA chain can offer a much more efficient way to find policies and other ways that can prevent attacks without technological changes to the software.

6.2 ABA risk or threat model

The author's first invented risk or threat model attempt is based on ideation is an attack library style approach that uses behaviour science to tries to prevent most typical security violations and vulnerabilities in specific context. It is combination of the ABA model, PASTA's attack patterns and the CAPEC attack library. The model requires building the library from scratch and continuous maintenance work.

The attack library should eventually be capable answer to the all types of threats that are typical for the business area. For instance mitigating the information disclosure style threats should be possible with the model since the most likely malicious behaviour is properly known and understood beforehand. Therefore prevention can be carried out in an innovative way just using antecedents, behaviour and consequences chain to describe good defence pattern for the attack. The attack library should apply as well to the difficult and time consuming attacks like sophisticated thefts, zero-day attacks and insider threats like the snooping done by the system developer. The defence patterns are built on real life cases so they should be best ways to mitigate similar crimes and threats in future. Most importantly, the root reasons behind the attack should be documented (Jackson 2012, 78).

6.3 The ABA attack library model description

The model requires a follow up of the security area news in a frequent basis, ABA analysis for the found news, building a defence pattern regarding to the news adding the data to the database and linking the threat to CAPEC's attack pattern when applicable. The database may contain predefined crime classes based on Table 11, how-

ever, the classification testing is currently something that requires first database implementation. The one idea was to link the ABA attack library to the crime classification manual list. However; since the most of the attacks are already well described and classified in other attack libraries, the one very useful idea towards the models would be usage of the proper attack library. According to Velez & Morana (2015, 462) the CAPEC attack library can be used for the own attack library and it is one of the most advanced attack libraries available. Therefore it could be most ideal to combine the CAPEC attack library XML within the ABA attack library model.

Table 13. The ABA attack library model prototype

Step	**Description**
Investigate the news archives and news about business area related crimes and threats as a regular basis. Write down the cybercrimes and crimes according to the business area and analyse them with ABA.	Investigate found cases from the news and analyse each case with: • Events and situations (antecedents). Basically the root reason or motive. • Actions made by individual or group (behaviour) • Consequences Handle the found cases and analyse antecedents (events and situations), actions and consequences based on news articles and any applicable source regarding to the case.
Define the defence pattern based on ABA analysis or use existing attack patterns from CAPEC.	Defence pattern should apply to business area or organization's needs. The pattern can be description how to apply policy, change to the physical security or adding the surveillance camera prevent the incident. When applicable it is good to use directly CAPEC's attack pattern.

Store and refine the data to the database of findings.	Store the events and situations, root reason, motive, actions and consequences. Classify the data based on following list: • Asset / target • Antecedents • Events and situations • Actions • Threat • Vulnerability / vulnerabilities • Defence pattern • CAPEC ID when applicable. • Impact
Use the ABA attack library for the software that is under planning, development or maintenance. Library scope can be applied to complex ensemble of security layers so not only for software development.	• Compare the business / organization specific ABA attack library incidents to product, product environment, processes and so on. • Use defence pattern to mitigate threats. • On software level use for instance STRIDE to recognize the threats as a supportive element.

The model applies the PASTA's stage six to identify the attack surface and enumerate the attack vectors on the storage phase. (Velez & Morana, 2015, 463).

6.4 Applying the new and former incidents

Figure 12 describes the main process of applying the new or former incidents for threats mitigation and it is graphical representation of Table 13. The basic idea is applicable for any organization or business area. The figure describes the searching, mapping the antecedents, actions and consequences and storing the data.

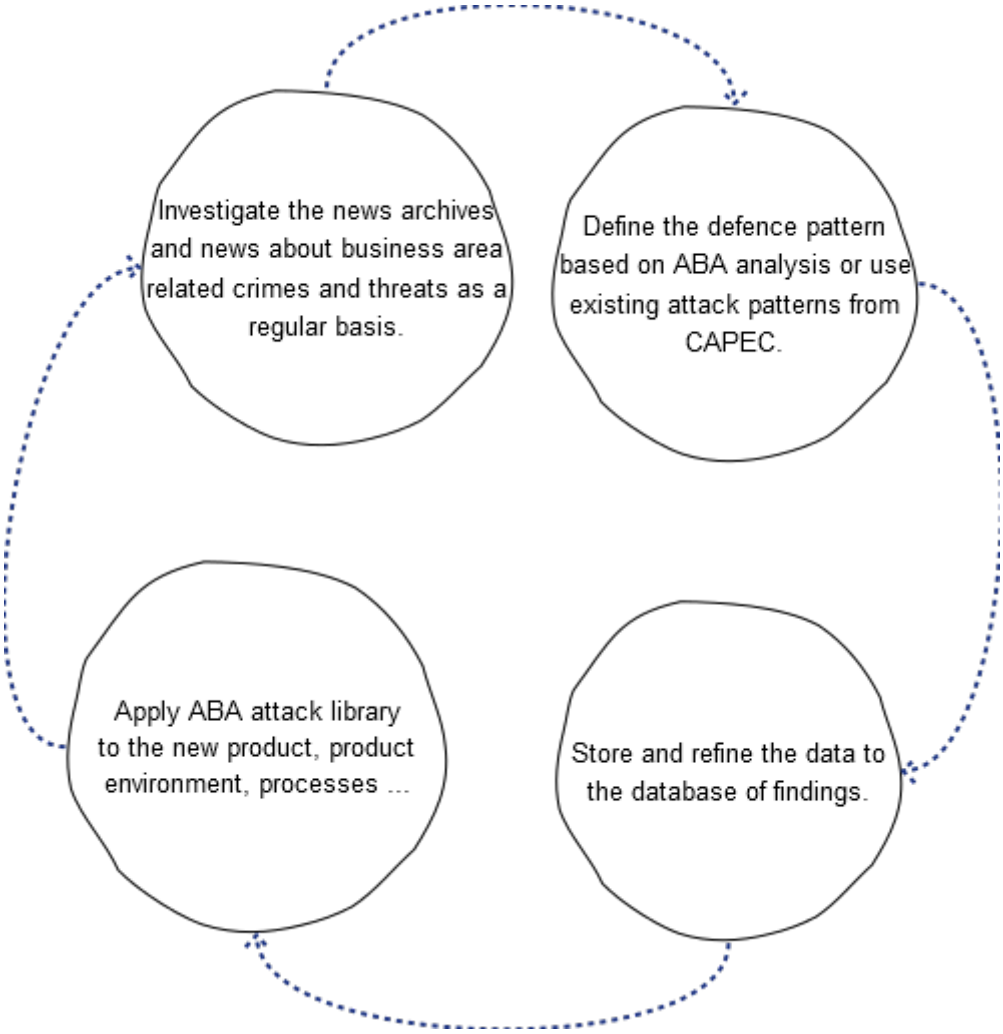

Figure 12. Applying the ABA on news archives to add a new entry to the library

The first step of the attack library model is to investigate news archives about the typical business or organization area crimes and apply ABA on them. Use of multiple sources when possible and investigation of the available background data from all available sources are recommended. Searching the same incident by other news

sources might reveal new data regarding to the case. The second step is to define defence pattern based on ABA analysis during first step. The events and situations can be also handled as motive or root reasons. The motive stands for what the attacker's aspirations and the needs to commit the crime are. The third step is to store and refine the data to the database. The fourth step to apply library findings for the wanted area to mitigate and defend for similar attacks in future. The loop should be continuous process to maintain the library during work.

6.5 Further ideas for the usage

The management can use the ABA attack library defence patterns to do decisions for instance for the product, environment or processes. For instance defence patterns data might be useful when selecting security criteria or ordering audits. The following figure illustrates the use case example for the management.

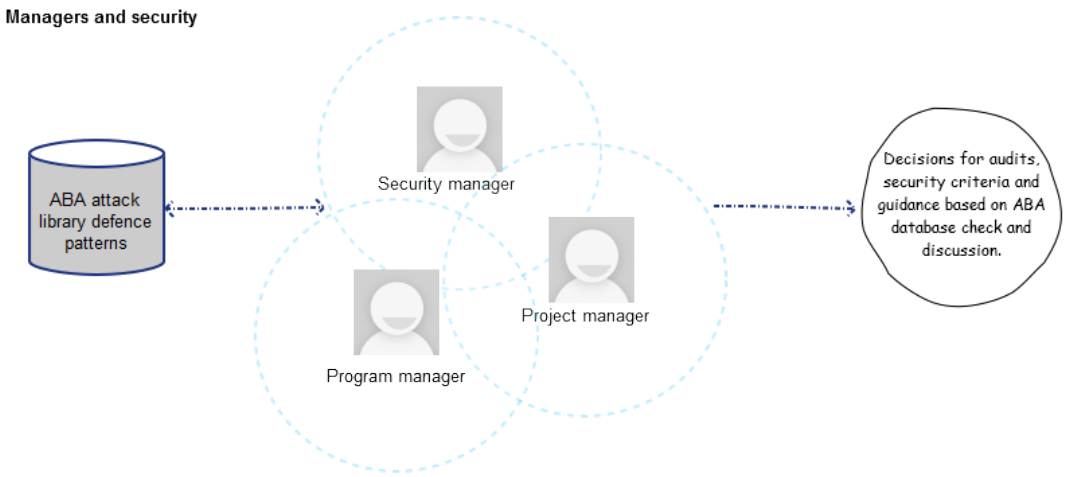

Figure 13. Managers and ABA attack library defence patterns

The agile development teams can use the ABA attack library to improve technical security based on the ABA database defence patterns. Figure 13 illustrates the use case example for the management.

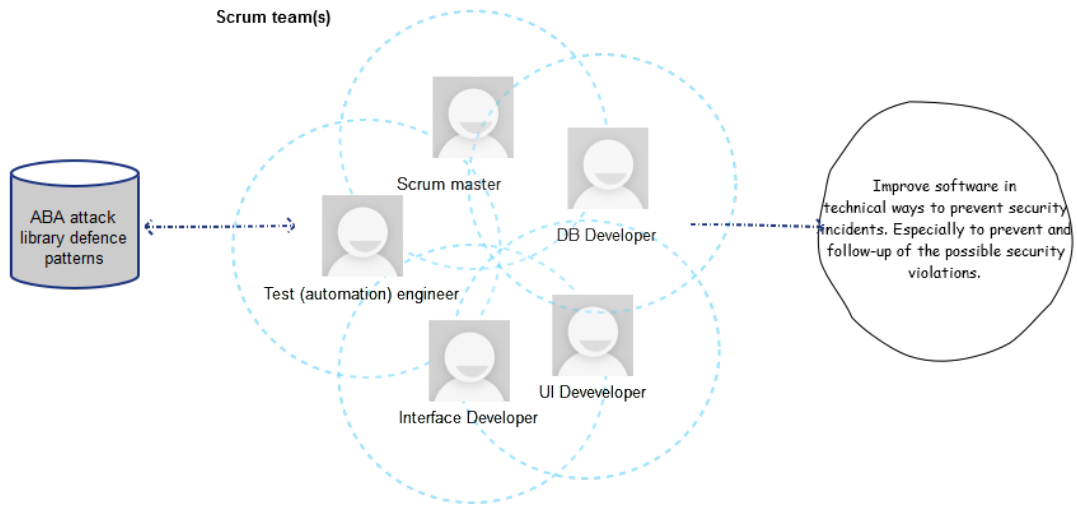

Figure 14. Scrum teams and ABA chain related security

6.6 ABA Attack library on Scrum

Figure 15 describes the usage of the attack library in the agile development model lifecycle. The figure does not contain the Scrum or Kanban lifecycle but instead of that gives a conceptual viewpoint of the attack library usage on this specific context.

The asset is the product or its part that the team develops within the agile methods. The first part is planning and requirements, the second is architecture and design, the third is development and implementation and the fourth phase is Testing and evaluation. The deployment is the last part and often done after one or more development rounds. The health care attack library is available in all phases; however, it is important especially for planning and requirements and architecture and design. Eventually the delivery will be done after testing and evaluation, and the product should be safer than without the attack library.

The attack library should be applicable on generic level to any agile model.

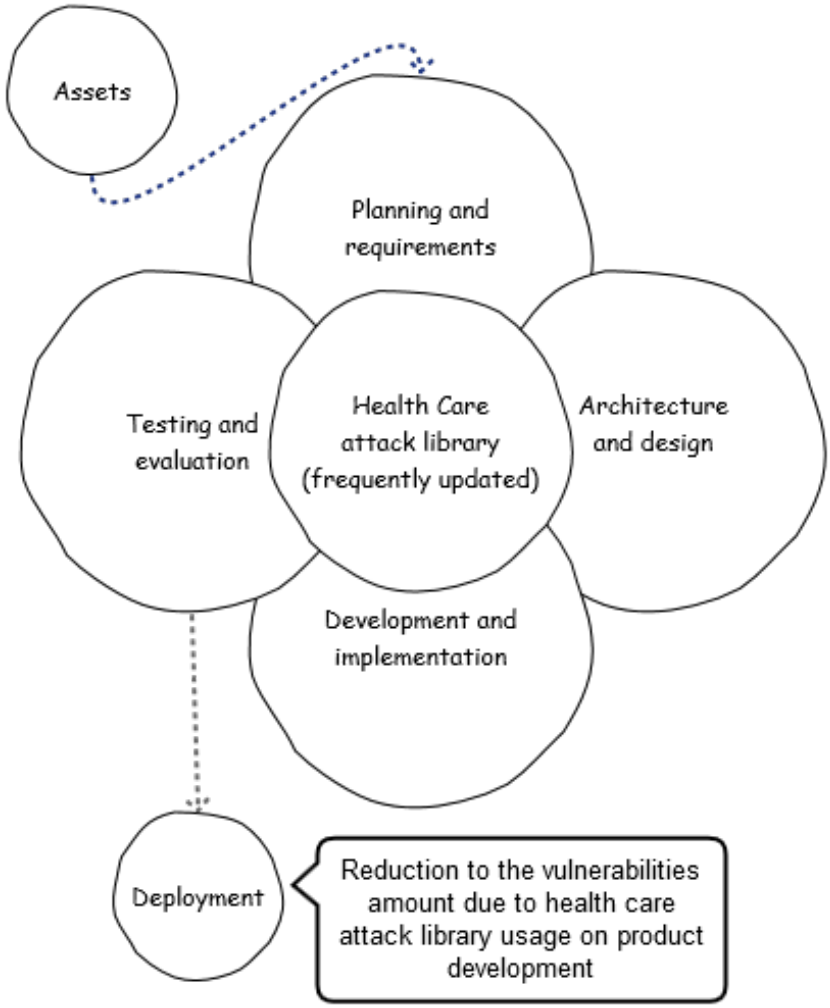

Figure 15. Health care specific attack library usage example on the software development projects (adapted from Koistinen, 2013, 93)

7 Research results based on questionnaire

After each workshop the attendees were instructed to write their answers onto the Questionnaire: Security risk and threat models. The questionnaire was sent to 19 persons and 15 persons answered to the questionnaire. Table 14 contains generic data of the teams which participated to the workshop. Also, the amount of evaluated systems and findings is listed on the table; however, unfortunately it is not allowed to include the findings to the thesis due to confidentiality issues. However; all STRIDE elements were used and the criticality of the findings was from low to critical. The evaluated systems were on proof-of-concept phase, development phase or under

planning phase, which definitely stands for that workshops were useful and provided meaningful results.

Table 14. The teams that performed the evaluation

Team	**Participants**	**Systems**	**Amount of findings**	**Site**	**Industry**
Development team	4	1	9	Jyväskylä, Finland	Industrial Internet
Development team	4	1	10	Linköping, Sweden	Industrial Internet
Architect team	3	2	16	Helsinki, Finland	Industrial Internet
Development team	6	1	11	Karlstad, Sweden (remote workshop)	Industrial Internet
Development team	2	1	6	Oulu, Finland	Healthcare
Total	19	6	52		

The first workshop was held in September 2015 due to summer vacation period and project schedules. The last workshop was held in the end of October at Oulu.

7.1 Basic data of questionnaire

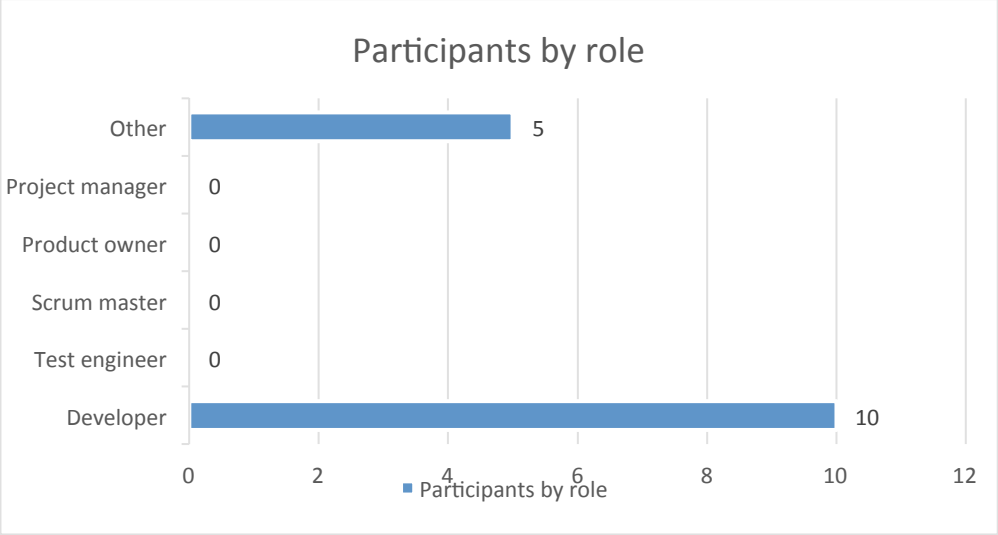

Figure 16. Participants by role

As seen on Figure 16, the classification of participants was not perfectly planned since the roles were not set as expected. A strange issue on the findings was that the teams did not have a Scrum master or the Scrum master did not participate at all. The author is the Scrum master for one team currently, therefore it explains the case from point of one team. The roles of the participants were mostly developers, architects, one chief architect, one software development coach and one "Sales, BD, Product owner". The author did not answer to the questionnaire at all since a personal view towards the models is definitely biased.

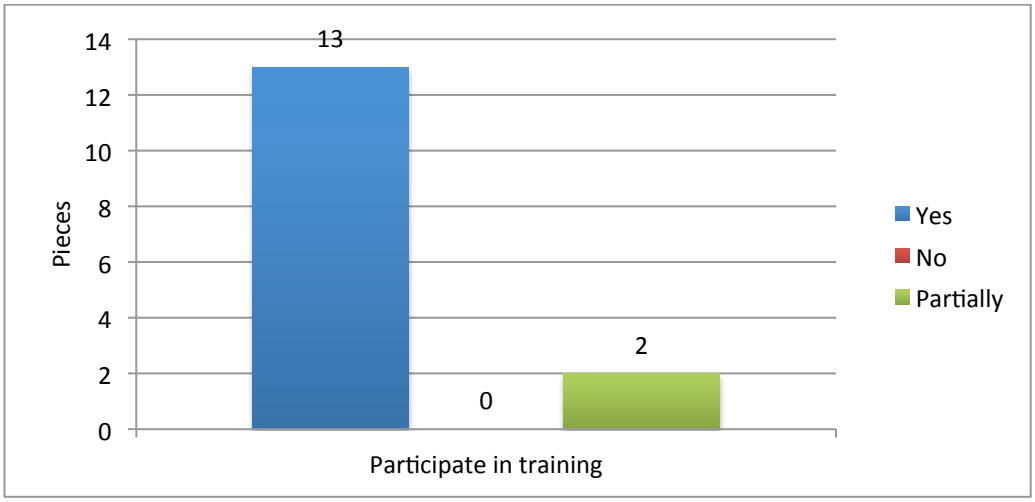

Figure 17. Participated in training

13 participants were on the trainings all the time. Only 2 participants reported that they joined to the training partially, which corresponds also to the author's view of workshop experiences.

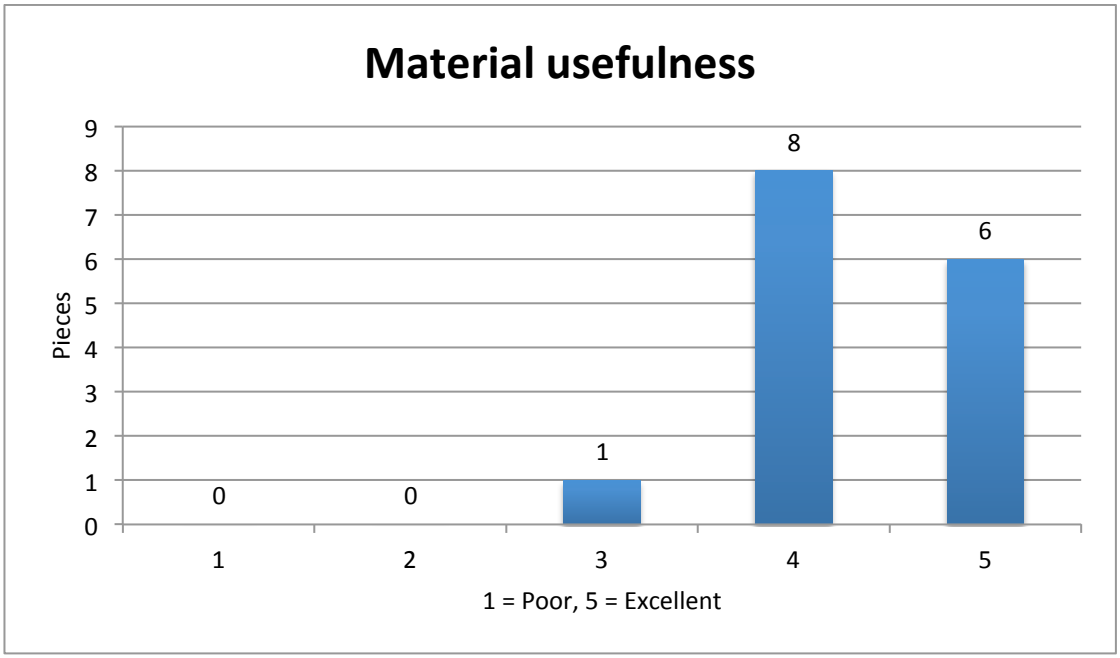

Figure 18. Does the used material support understanding of the selected threat models?

The material was seen as a supportive element for training, and small improvements or tweaks were made after each workshop, however, on written feedback a comment was received that the examples could be more vivid. Also, one workshop was held remotely and in future the author's all workshops will be held locally. Based on discussions with participants more practical examples could be useful in future. The author shares the view and for best possible training each STRIDE element should be represented with video or tangible examples drawn on a whiteboard. Generally speaking the PowerPoint presentation, Elevation of Privilege game and YouTube example videos were probably fine, however, CAPEC needs gamification style approach or something else. Just linking the CAPEC findings to the found issues is not the best way.

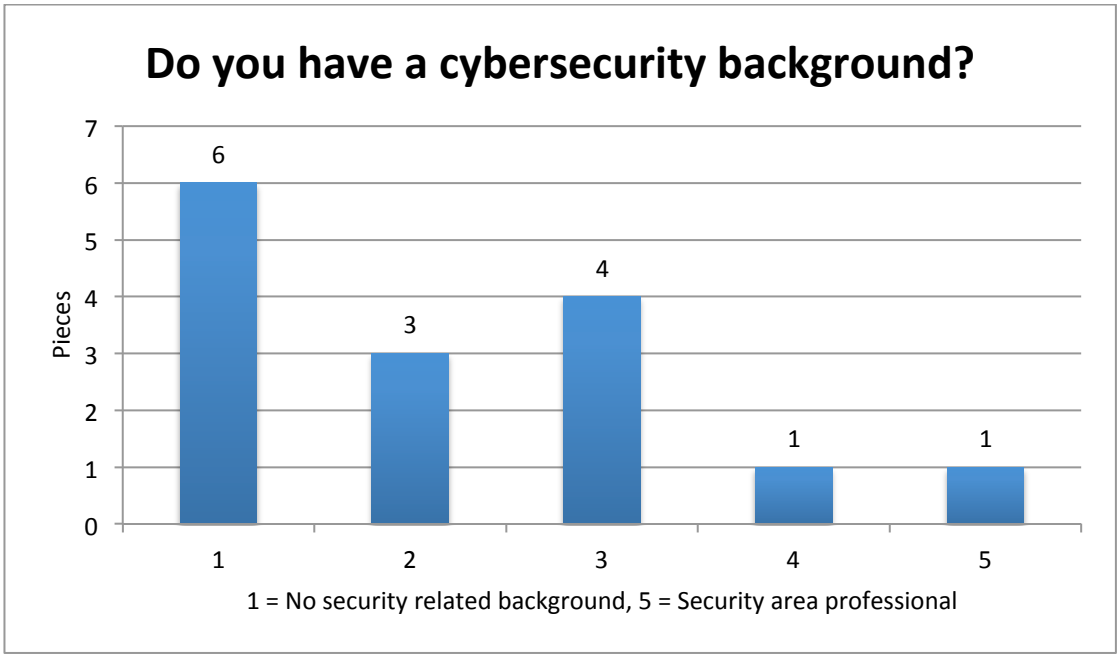

Figure 19. Do you have a cybersecurity background?

The diversity of the answers was surprising since it was expected that people do not have cyber security knowledge at all; however, four participants considered that they had moderate knowledge and two respondents evaluated knowledge on higher levels where the knowledge is expected to be very good. Nevertheless; from the point of security training the company should arrange security related training much more to tackle down the growing needs of security.

7.2 STRIDE

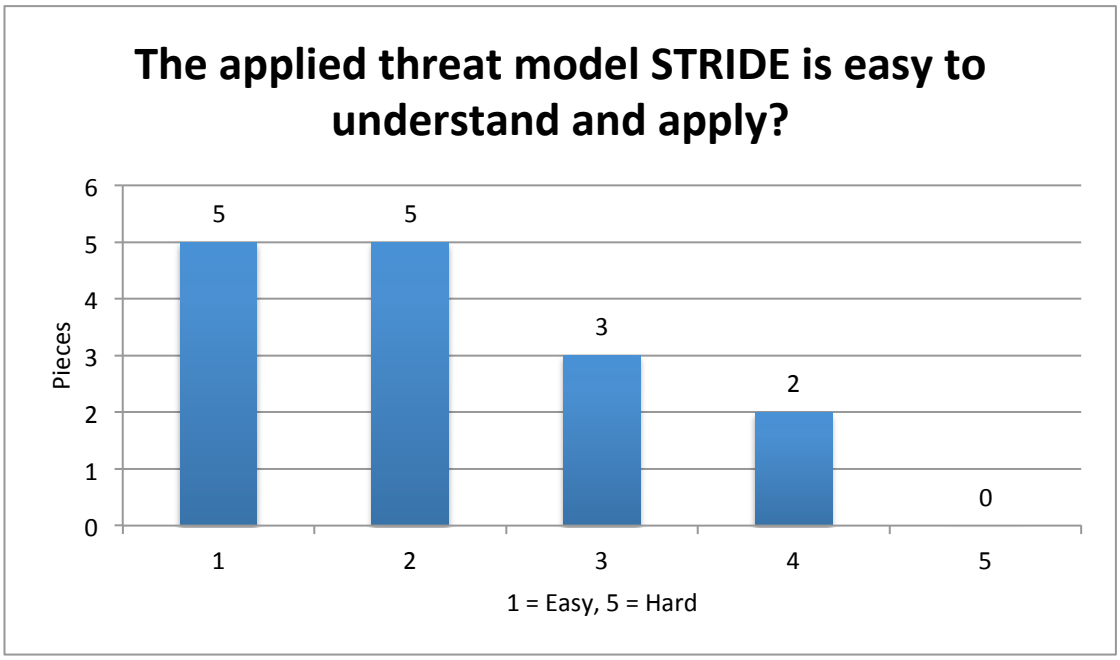

Figure 20. The applied threat model STRIDE is easy to understand and apply?

It was found out afterwards that the scale of the answers was set in the wrong order. One in the scale meant easy and 5 meant hard, which was explained to the participants after one participant commented on the issue. 66% of answers evaluated STRIDE as easy to understand and apply with the majority of answers on values one and two. Three answers were set to three, which means moderate and one to four that stands for quite hard. Root reason is very difficult to evaluate; however, at first the workshop was not fully successful since it suffered from time problems and other small drawbacks.

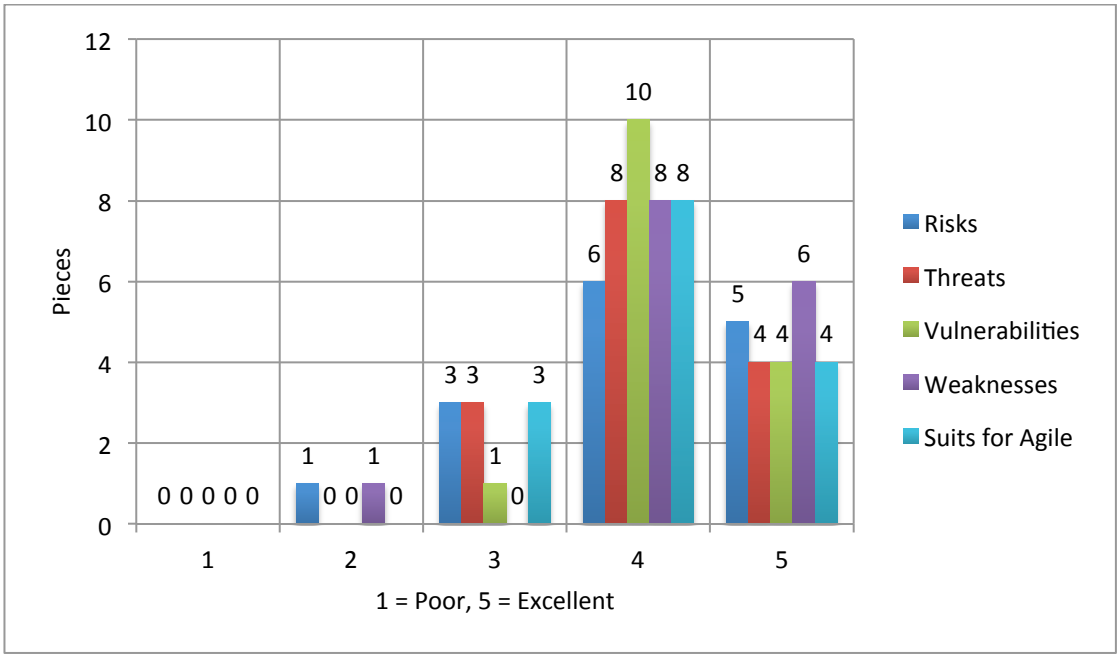

Figure 21. STRIDE support to recognize risks, threats, vulnerabilities and weaknesses

On the scale one meant poor and five meant excellent. Based on the answers the majority of the answers was set to very good (four) or excellent (five). Based on feedback most of the answers evaluate STRIDE as very good or excellent for recognizing the risks. In the planning phase it seemed a good idea to ask about the suitability for risks, threats, vulnerabilities, weaknesses and agile; however, since most of answers are very good or excellent for all of them, there must have been misunderstanding between the terms. In the author's opinion STRIDE is not a good model for risk evaluation since it is not a risk model. It seems that 4 participants understood that STRIDE is not a good risk model. Clearly the problem was that the presentations did not contain definitions of risks, threats, vulnerabilities and weaknesses and that is definitely design flaw on the workshops. As a summary it seems that STRIDE got good admittance from the teams.

7.2.1 Stride advantages

Based on the feedback most of the answers evaluate STRIDE as a very good or excellent model for better security. The advantages based on feedback were that the card game was an easy approach, and that helps to raise a discussion regarding to the system security. The discussion was actually very good in all teams and generally

people are really interested in security issues. One feedback was about discovering threats in an existing product, and indeed, the model works on planning table but also on the evaluation of existing systems.

The model was considered lightweight and Microsoft's tool support was considered a benefit. The usage of the model was possible without long training sessions for developers. The gamification approach was successful so the author can definitely recommend it based on results.

7.2.2 Stride disadvantages

STRIDE was considered a high level model and it was suspected that more competence for detailed analysis was needed. Also, finding solutions to threats was considered to be out of the focus in this model. Definitely one of the problems was the lack of time since the workshop was only one day long for most of the teams.

The model needs complementary support from other models like CAPEC or processes, and the testing viewpoint was missing as well. One comment was about the too diverse discussions, and the card game rules also need improvements. As a result the card game rules were updated for this thesis. At the first time the game consumes easily 4 hours.

7.3 CAPEC

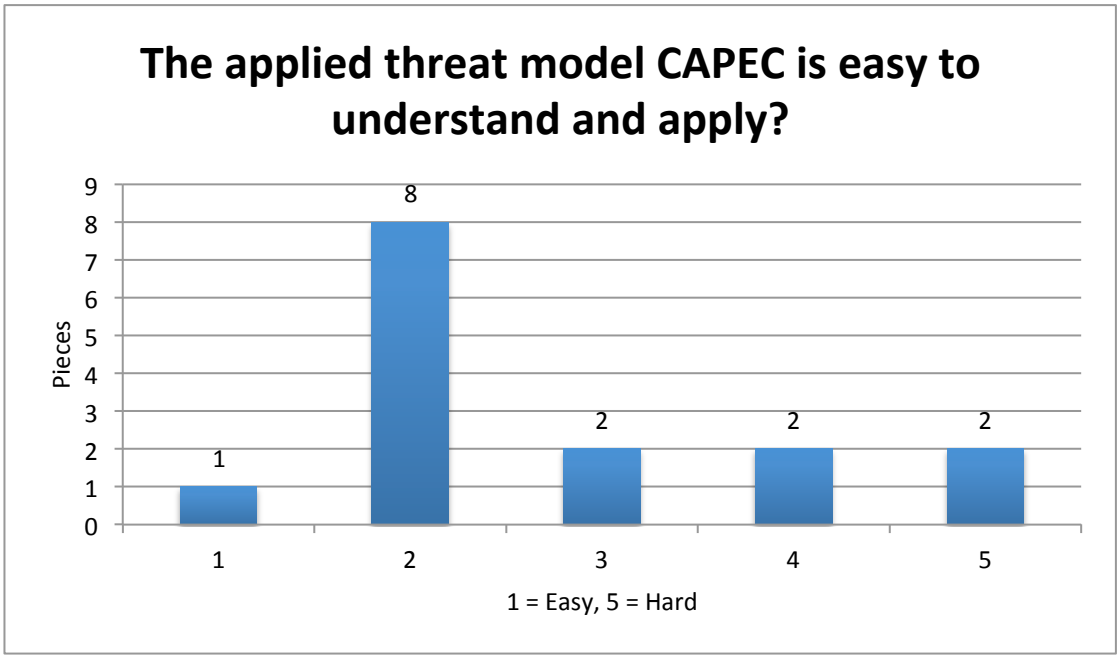

Figure 22. The applied threat model CAPEC is easy to understand and apply?

The scale was one (easy) to five (hard). The blue bar defines how easy the model was to understand and 9 persons evaluated it as easy or quite easy; however, especially on written feedback the model was evaluated hard and boring.

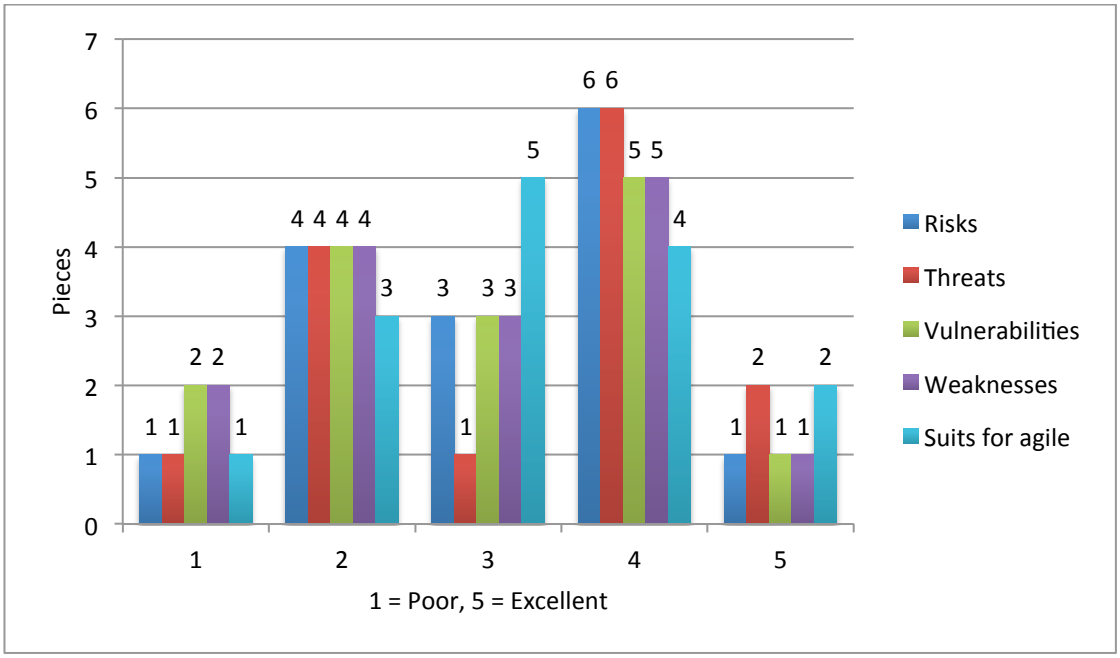

Figure 23. CAPEC support to recognize risks, threats, vulnerabilities and weaknesses

CAPEC part on the workshop was something that caused a wide diversity of answers, which suffers the same problem as STRIDE's answer set. The participants do not necessarily understand the difference of risks, threats, vulnerabilities and weaknesses since those terms were not explained in the presentation. Criticism is understandable since the STRIDE's gamification approach was more entertaining than CAPEC's boring "search and map the correct attack pattern to the findings with Google search or domain based search". Still, on many comments it was seen that it is a good supportive element for STRIDE. Generally speaking the problem was the CAPEC part in the workshop, not the model itself. One interesting viewpoint on the CAPEC was that eleven answers of fifteen answers were good, very good or excellent for agile compatibility. Basically what this stands for is that CAPEC works in agile development when it is correctly understood and used.

On results analyzation it was found out that PowerPoint presentations did not include deeper CAPEC description since many patterns consisted of several elements e.g. summary, attack execution flow, attack prerequisites, typical severity, resources required, related weaknesses and related to patterns. That was a flaw on the workshop materials since additional elements like severity helps to prioritize the found threat.

7.3.1 CAPEC advantages

CAPEC caused a great deal of confused feedback; however, in many feedbacks it was seen that a big attack pattern library with clearly explained patterns was a benefit. Some participants considered that searching was easy and worked right away. The structure of the library was also seen as good. One benefit was to reproduce recognized issues and apply suggested solutions to recognized issues. Cooperative usage with STRIDE was seen as a benefit. Based on the answers CAPEC fits for agile as a supportive element and is a working pair for STRIDE still in need of more evaluation.

7.3.2 CAPEC disadvantages

In the feedbacks CAPEC was mentioned as a large, complex, exhaustive, boring and heavy to read database. Prioritization finding with CAPEC was seen as difficult; however, it is not necessarily the library's task to prioritize issues as OWASP TOP 10. Based on one feedback CAPEC has no reasonable way to use and never should be used as an only model. The clear drawback of CAPEC was that it is very technology focused but still new technologies were not considered implicitly. Lack of discussion was seen as a problem as well.

7.4 Improvement proposals regarding to the threat models usage

Feedback contained many good improvement proposals for the future threat modelling workshops and more secure software development. The workshops were seen as a good starting point for security improvements. One direct and fully doable improvement idea was to have thesaurus for mapping the EoP card deck cards descriptions directly to CAPEC's attack patterns in order to enhance testability. For instance "*6 Repudiation normally finds CAPEC-<nn>, CAPEC-<mm> & CAPEC-<ii>...*".

In feedback the frequency of workshops was raised as an important issue as well as adding threat modelling as part of typical software development cycle during sprints. One participant also requested that teams should have a different threat model in workshops as a frequent basis, which could be a good idea; however, only few models might fit the agile development such as TARA, DREAD and OWASP TOP-10 but definitely they will be taken into account also in the future.

Prioritization and defining the highest risks were seen very important and that is definitely true. The selected models did not take the risk evaluation into account so afterwards it is easy to say that TARA or DREAD would be a good answer for the highest risk addressing. Also, the big picture handling with whiteboard planning sessions, defining the trust boundaries analysis as a whole in combination with other STRIDE tools should be taken into account on threat modelling training.

To summarize: the most important feedback was to add more practicality and more than one workshop, give for instance tool related training for known models, do threat modelling frequently as a part of sprints and take highest risks and trust boundaries into account.

7.5 Improvement proposals regarding to training

Clearly one most wanted improvement was that more than one workshop was needed, which means the cross-sectional approach for the thesis was a wrong solution. Basically a better approach would be longitudinal since the workshops should be held at least 2 or 3 times for same area and the models evaluated after each session. Also, there is no sense to have too short workshops so one day timeframe should be reasonable also in the future.

The original EoP rules from the card deck were seen problematic so the new instructions were updated based on the feedback on this thesis. Also, well planned tutorial based on Microsoft Threat Modelling tool was seen as an important part and it was missing from the contents of the original workshop.

The presentations were generally good; however, continuous improvements are needed. Threats should be formatted into scenarios, and the risk modelling should be included in the workshops. Adding DREAD or TARA to future workshops could be needed improvement. Basically better examples and more concrete explanations for the terms are essential in future workshops.

8 Threat models evaluation based on literature

The threat model evaluation is done primarily based on literature analysis but replenished with the data from Internet's electronic sources. For instance, TRIKE or DESIST were an examples of a models which does not have the easily available literature describing them. The secondary evaluation is done with a self-administered questionnaire. The threat model evaluation is supported by multiple discussions with

Tieto's representative; however, the discussions may lead to biased view towards the model selection.

According to held discussions with Tieto's representative, the most suitable threat model would be:

- It should answer to the health care software development needs
- Usage of the threat model should not consume too much time
- It should not be too laborious
- It should be easy and fun to learn
- It should reveal wide range of the security threats with small effort
- It should work well together with Microsoft products

8.1 Brainstorming evaluation

Whenever a quick and light model is needed, the brainstorming methods might be useful; however, threats might be difficult or even impossible to point out and as a method it requires removal of borders and scope. On the other hand, threats might be very dependent on the participants and facilitation of the brainstorming meeting. Definition of the exit criteria may be difficult and it is even more difficult if an exit criterion is completely missing. (Shostack, 2014, 31-34). In the author's view brainstorming is not an answer for the healthcare security problems. It is simply a model too lightweight and does not contain any kind of security approach.

8.2 DREAD evaluation

The model was meant for risk modelling; however, the model is not a good choice anymore since the results are not useful in all cases. (Shostack 2014, 180). Microsoft's SDL team will not recommend the usage of DREAD anymore. Therefore, the model can be considered as obsolete. (Shostack 2014, 180) and it is not selected for the further evaluation. In the author's view DREAD looks simple and easy to implement. It could be an easy stepping stone towards security; however, due to its obsoleted status it is not a choice for further investigation or evaluation. DREAD was very

interesting model since it answered to risk evaluation whereas STRIDE was more about recognizing threats.

8.3 STRIDE evaluation

The gamification approach was definitely successful for teaching STRIDE.
In the author's view STRIDE was a very promising model. It has few clear benefits; however, it also contains certain disadvantages.

Advantages are listed below based on literature analysis and questionnaire:

- Fits for agile development cycles on development and architectural levels
- The evaluation of privilege card game offers easy stepping stones for any developer. Gamification is the key to fun learning.
- Actively supported and used by Microsoft
- Free Microsoft Threat Modelling Tool available for more serious usage.
- Basically fits for any software, not only for Microsoft's products
- Does not prevent usage of other threat models like the attack libraries
- Found findings are possible to report as typical bug reports, feature requests or other tasks

Disadvantages:

- STRIDE is high level model and might need more competence for detailed analysis
- No risk based approach or prioritizing the threats, needs another model or approach to resolve prioritization and risks handling
- Does not take technical defences into account
- CAPEC compatibility not best possible.

8.4 TARA – Threat Agent Risk Assessment evaluation

TARA was the last found security risk model from Internet in September 2015. The model came up by coincidence when added the missing references to the thesis. Based on Intel's material noticed that model applies quite well to healthcare area risk filtering but not necessarily for agile development. The main problem with TARA was finding the literature regarding to the model. However; all needed material is on Intel's website.

Based on Intel's whitepaper the model's advantage is to evaluate big amount of risks, put them in understandable form on big picture and communicate findings to further decision making. Disadvantage is that model does not take impact of the risk into account. The model might be too heavy for typical agile development team but still useful for architect teams or the specific security assessment team. However; the model needs practical testing before understanding it more deeply and can evaluate it for healthcare purposes.

The model supports building its own threat agent library for business area purposes and basically it should not be overly difficult or time consuming job. The model is recommended be taken under further evaluation if the organization needs the model for risk evaluation in critical projects. Definitely worth of thesis.

8.5 PASTA evaluation

The first impression of the model expresses a throughout structured and sturdy model for big organizations that need to fight against new type of threats; however, the model consumes resources heavily because it has seven main stages and each of them have many demanding requirements. A vast amount of people from different roles and organization levels in the organization is required to participate. Eventually, the system has to be decomposed to components, specific data flow diagrams and use cases if not done before. Basically, a reasonable PASTA usage requires implementing all seven stages within company's processes and that requires a very mature organization and a very strong process culture. The model is not taken to further

evaluation simply because it consumes too much resources and easily approachable model was very important; however, it is one of the strongest and newest models available in the field, thus, whenever the security must be on very high levels the organization should consider it. The author's opinion is that the PASTA would be one of the most secure choices for the healthcare software development.

8.6 TRIKE evaluation

The author found that TRIKE contains useful elements like requirements and implementation model. However; the risk model is experimental, attack trees are deprecated and the model has multiple layers that may require heavy resource investment. The major problem within all three TRIKE versions is that each of them is unfinished based on found data. (Larcom & Saitta 2012). The TRIKE's website is outdated and not updated regularly. (Larcom 2012). The first version has status DRAFT, the second version 1.5 status is unclear, and the third version 2 is under development. Hence, the threat model is inadequate for production environments but the license model supports the further development based on MIT licence. Basically, usage of TRIKE requires Excel's spreadsheets or the standalone tool, however, it is possible to use it with traditional pen and paper approach. The new tool for TRIKE version 2 is under construction. TRIKE is still under development; however, the last update to the home pages was three years ago, therefore, the project state is unclear to the author. (Larcom & Saitta 2012, Tools).

The author tried the tool and the spreadsheet to find out their usability for health care product development purposes. It just turned out that the standalone tool and Excel spreadsheets were likewise unfinished. The standalone tool usability is simply cumbersome and it lacks intuitive look and feel. According to the author, the model may be a challenger for other threat models in future; however, the threat taxonomy is very limited as it covers only elevation of privilege and denial of service attacks (Saitta, Larcom & Eddington 2005, 8). Definitely TRIKE requires a vast amount of resources, planning and training before the model could be a part of the organization's daily work since it requires overall communication between stakeholders and security teams.

According to (Saitta, Larcom & Eddington 2005, 14), TRIKE fits for waterfall and agile style development; however, the author's opinion is that model fits easier on waterfall style development cycles. The model expression is unfinished and therefore it is recommended to wait for the version 2 before further evaluation. In future TRIKE may be a strong challenger.

8.7 OWASP TOP-10

OWASP TOP-10 project is an attack library and very popular choice on web applications area. However; it offers only 10 most exploited security vulnerabilities towards web applications and gives biased view towards security. Usage of it is useful for example with STRIDE since they replenish each other. However; CAPEC offers much wider attack pattern library and is professionally maintained. OWASP has professional support; however, it leans heavily towards open source community and participants do not have specific requirements or qualification. The model is gives first stepping stones towards threat models from the attack library point of view. The model is not recommended as only model towards health care but can be used in the beginning to understand most typical threats for the web applications.

Advantages

- Easy to learn.
- OWASP offers documentation how to fix these security flaws.

Disadvantages

- It is meant for web applications only.
- It is an open source community's product.
- The TOP-10 list may lead to falsified understanding and denial of other security flaws.

8.8 CAPEC evaluation

The Common Attack Pattern Enumeration and Classification attack library is one of most comprehensive attack libraries available. It contains attack patterns for many different domains and each attack is documented with the ID and clear description. The library was clearly the best of the found attack libraries and contains over 460 different attack patterns. Since nowadays programming is based on patterns and object oriented programming, the CAPEC approach is definitely up to date. However; latest technology is not covered on CAPEC and it is considered as negative viewpoint.

Advantages:

- Comprehensive (460+ attack patterns)
- Answers technical questions whereas STRIDE cannot do that
- Clear predefined library structures
- Typically severity of attacks described
- Patterns contains many helpful descriptions like summary, attack execution flow, attack prerequisites, typical severity, resources required, related weaknesses and related to patterns.

Disadvantages:

- Not gamification approach. Feels complex or boring for many users
- New technologies are not represented as much as possible
- No clear risk based approach

8.9 Evaluation of the ABA attack library and the agile threat model future development

As a basis for ABA attack library evaluation its ideation and cancellation was done in spring 2015 without knowledge of the Intel's threat agent risk model and without the reasonable workload estimation that such library development requires. According to Shostack (2014, 31-34), "Developing a new library requires a very large time investment, which is probably part of why there are so few of them." That is also the

main reason why the development was cancelled. It simply consumes too much time and resources.

The cancellation decision was made referring to the discussion with Tieto's representative, such a model is too laborious as it binds at minimum one person for researching threats to the new attack library development. Additionally, based on the author's own estimation a new model needs 2 - 4 years planning, evaluation and testing with continuous improvements. In this case getting resources for new library development was not an option so that research path was abandoned and model is the first version without any specific improvements. To summarize practically, the model needs more development and evaluation in a similar way as any existing main risk and threat models. Additionally, combining the new threat model on agile area and development of the behaviour based threat model for software development is clearly worth a separate thesis or even a doctoral dissertation. In this way it is possible to collect feedback and update or redefine the model. It is important to notice that the unbiased evaluation is not possible without longitudinal testing, science community's feedback and iteration.

The basic concept of the model is simple: investigate news, define the defence pattern for the incident, store and refine the data to the database and eventually apply library in reality. The model's strength is assumed to select correct defences to prevent and mitigate the threats in the best possible way based on existing knowledge of old threats and incidents. A certain weakness is that the model is not tested in real life and may contain design flaws due to too straightforward thinking. The current attack library model needs also STRIDE or another threat model as a support since the attack library does not take interfaces and attack vectors into account. One good risk based addition could be Intel's TARA model.

9 Conclusions

9.1 Conclusions on research questions

What appropriate risk and security models already exist?

Based on research appropriate models are STRIDE, DREAD, DESIST, CAPEC, OWASP TOP-10, WASC Threat Classification, TRIKE, P.A.S.T.A and TARA. Project management risk methodologies were not taken under evaluation but project risk evaluation was used in one workshop after applying STRIDE and CAPEC. Basically also project risk evaluation methods apply as a supplemental method to the existing thread models. However; project risk methods handling was not a part of the thesis scope. Each method has its advantages and disadvantages but the selection of the models for further evaluation was done based on agile development frameworks, literature analysis, existing models' evaluation and based on discussions on author's and healthcare representative status meetings. The own ABA attack library model was result of research done in spring 2015.

Which existing risk and security model fits best for health care software development processes?

Based on literature research and evaluation, suitable models for agile development were STRIDE, CAPEC, OWASP Top-10 and TARA; however, Intel's TARA was found in a very late phase and thus was not taken into further investigation. STRIDE and CAPEC were selected to further evaluation since they were the most promising at the time. Based on questionnaire results from the teams STRIDE is a very good model for agile, and the training method was the right choice for most of participants. However; CAPEC caused some difficulties due to a wrong training approach; however, it definitely helps to understand the attack patterns that can be used for STRIDE findings. The gamification style training of the STRIDE and practical examples are good stepping stones towards more serious threat modelling.

P.A.S.T.A was probably the most comprehensive threat model and maybe suitable for healthcare purposes; however, fitting it to the agile development models and the

organizations existing processes was considered too difficult and time consuming. Nevertheless, it is definitely one of the models that should be taken under further evaluation and is probably good choice for Master's degree thesis topic. Likewise Intel's TARA is worth of further evaluation. The own ABA based attack library is an attempt to develop a new model for healthcare purposes but the development in this thesis was cancelled in May 2015.

What are the weaknesses of existing models?

The following table contains the found weaknesses of the models based on literature and evaluation by development teams.

Model	**Weaknesses**
STRIDE	High level model and no viewpoint towards vulnerabilities handling. No risk evaluation and does not always offer solutions for threats. Available gamification approach good for learning the model but not recommended for fully organized approach. Using of the Threat modelling tool is recommended. Does not include technical defences.
DREAD	Already obsolete model. According to literature review may lead to weird results.
DESIST	Too similar and no proven benefits over STRIDE. Found only one reference and two S letters on acronym is confusing.
CAPEC	Attack library is clearly difficult to promote for development teams since similar viewpoint of viewpoint of gamification was missing. Not suitable as an only threat model.
OWASP TOP-10	Only for web applications and contains only ten most common vulnerabilities. Definitely not recommended as an only threat model.
WASC Threat Classification	Obsoleted. Attack library latest version released 2010.

TRIKE	Model still under construction and no literature available. Not ready enough for serious development usage.
P.A.S.T.A	Does not respond needs of agile development teams. Difficult to include on Healthcare area processes without a great deal of work.
TARA	Does not take risk impact into account. Might be too heavy model for agile development teams. However; practical testing of the model is recommended.
Brainstorming methods	Does not offer sensible security viewpoint.
ABA attack library model	Only initial version created without practical approach, testing and iterations. No proof that model would work in reality.

Does the developed health care specific risk and threat model offer better risk and security management than generic models?

Eventually the author ended up to use the existing models STRIDE and CAPEC since the developed ABA attack library testing and building would require a vast amount of work and resources. Since there are no practical evaluations done for the ABA attack library, the developed model does not currently offer better risk and security management than commonly available threat models; however, it is possible to take the ABA attack library model under further iterations and development under another study. The author believes that simple and easily understandable threat models are key to better security in future, and complex threat models are rarely used.

9.2 Summary

The objective of this thesis was to study and explore a variety of information security risk and threat models and select or develop the appropriate model for healthcare unit's needs of the domain.

The result of the research project was definitely positive and useful for Tieto's Healthcare and Industrial Internet units since instead of one domain now two domains at Tieto can use selected threat models to improve their security in a meaningful way. Basically the tested threat models should fit to any organization in Tieto which follows typical agile development models. They should fit to other style development workflows as well, however, this thesis concentrated only on models that fit the agile methods.

As a concrete result with STRIDE teams fifty-two findings on five separate workshops were found. Most of the findings were mapped to CAPEC attack patterns. The big amount of findings reveals that taking security viewpoints into account with threat modelling in design phase is extremely important; however, it is important to remember that threat modelling is only one part in the whole security processes. The evaluated systems were on different development phases and some of them are going to production in 2016.

The thesis handled multiple different threat models like STRIDE, DREAD, DESIST, CAPEC, OWASP TOP-10, WASC Threat Classification, TRIKE, P.A.S.T.A and TARA. Also, Microsoft's SDL and brainstorming methods were under evaluation. Additionally, a new model was invented based on behaviour sciences area and certain existing models. The author's own model was personally the most interesting; however, the model was abandoned due to too high work amount to develop the practical implementation. In the research problem chapter it was mentioned that "Eventually the goal is to develop a new model based on the best parts of existing models"; however, that was not the case. Actually, the existing models were good enough and they met the typical agile development team's needs. Cooperation with STRIDE and CAPEC offered very good results.

Based on literature analysis few models were selected as applicants for practical testing on the workshops. The models were STRIDE, PASTA, OWASP TOP-10 and CAPEC. Eventually STRIDE and CAPEC were selected since they looked easy to learn and should not consume too much precious development time. Also, STRIDE's gamification approach was a clear benefit but it was unclear where and who has used the

gamification approach for threat model training in reality. OWASP TOP-10 looked good at the first glance; however, it contains only web application threats and only 10 most used attack vectors. It was decided to ignore other OWASP projects since open source without literature is very difficult to evaluate scientifically. CAPEC was selected since its usage is not difficult and its attack patterns library is one of the strongest available. Intel's TARA was found too late and therefore was not part of the evaluated models.

Originally the intention was to use software development teams shadowing to collect data of threat models usage; however, practical limitations such as too big expenses to the company declined that approach. The selected multiple choice questions are not the best approach for this kind of research project but combining it with the wide literature research made evaluation possible. Afterwards it is also easy to see that instead of cross-sectional time horizon longitudinal would be more reliable for evaluation. Workshops with STRIDE's gamification approach were considered successful and a good way to give a good start to the threat models world; however, on the other hand the second part of the workshop was often considered as boring and difficult because there was not the gamification approach or vivid examples.

The selected research philosophies and approaches were applicable and the best choices for the research within the practical limitations; however, the time horizon should definitely be longitudinal. Cross-sectional approach gives only a preliminary idea - with the longitudinal approach the research could be more reliable.

Afterwards it is easy to see that the fourth research question was too extensive since it required the creation of author's own model instead of evaluating only existing threat models. An attempt to create one's own ABA model based on behaviour sciences was too ambitious in the planned timeframe. Actually, about four months of research time were spent for that specific area; however, it offered a new viewpoint towards threat modelling that is not the typical way to handle the system related threats. Behaviour science, attack libraries and agile models were part of that ideation and research. Areas where further investigation could be conducted could be

e.g. development of a simpler model to prevent espionage and terrorism on software development area.

9.3 Combined effort for Healthcare and Industrial Internet units

The original intention was to conduct the research for Tieto's healthcare unit only. However; Tieto's Telecom unit's co-operating negotiations in autumn 2014 opened a possibility to move on new organization. Eventually the author started in Tieto's Industrial internet unit in January 2015 and agreed to continue the thesis few weeks after. Industrial Internet had a similar need for the threat models so it was decided to work for both units and cancellation of the ongoing thesis was not an option anymore without losing trust.

Originally the intention was to take the healthcare teams to practical tests in autumn 2014 – spring 2015. Working on the thesis started in July 2015; however, the thesis development was slow and the author concentrated on finishing the main studies in autumn 2014. Most of the work has been done in April 2015 – November 2015 and mainly off work hours despite of Industrial Internet unit's promise to use available working hours. In reality there was very little time to write the thesis during worktime.

In the end one healthcare team and four Industrial Internet teams participated in practical tests due to the tight timetable problems of the healthcare unit. Three Industrial Internet teams were dropped due to timetable and healthcare unit had tight timetable on autumn 2015 so that is why only one healthcare unit participated. The tests were executed with Swedish and Finnish teams and eventually combined efforts really made it possible to achieve good results with the thesis. The total calendar time used for thesis was sixteen months starting from July 2014 and ending in November 2015. The longest break on the development was in autumn 2014.

9.4 Further research

Research revealed several possible research paths. Following topics could be considered in future:

1. One possible topic is the ABA attack library and further development or new ideation and evaluation of the model. It may apply also to higher studying levels. Behaviour and threats can be definitely combined to a simple and easily usable model; however, that needs more ideation, evaluation and practical testing before it could be formed as a fully working threat model instead of simple attack library.
2. Apply gamification and threat models to improve learning processes and happiness of developers. Idea would be develop rewarding and interesting training based on gamification theories for software development teams.
3. Practical research project of P.A.S.T.A or Microsoft's SDL to the big organizations own processes.
4. Develop practical work instructions for developers of the latest version of Microsoft threat modelling tool.
5. Single thesis idea is to adapt TARA – Threat Agent Risk Assessment to health care purposes and build own Threat Agent Library and practical usage instructions.

REFERENCES

Carr, J. 2000. *Handbook of Applied Behavior Analysis*. ProQuest ebrary.

Casey, T. 2007. *Threat Agent Library Helps Identify Information Security Risks*. Accessed on 5th November 2014. Retrieved from https://communities.intel.com/docs/DOC-1151

Cole, E. 2013. *Advanced Persistent Threat - Understanding the Danger and How to Protect Your Organization*. Waltham, USA: Syngress.

Douglas, J.E., Burgess, A.W, Burgess, A.G., & Ressler, R.K. *Crime Classification Manual, A Standard System for Investigating and Classifying Violent Crimes*. 2nd ed. San Francisco: Jossey-Bass.

Doagu, S. 2012. *Työntekijä jäi kiinni väärinkäytöksestä: Tonki yli 100 potilaan tietoja luvatta [The employee was caught for abuse: Dug up for more than 100 patients' data without permission]*. Page on Keskisuomalainen 26th September 2012. Accessed on 24 May 2015. Retrieved from http://www.ksml.fi/uutiset/kotimaa/tyontekija-tonki-yli-100-potilaan-tietoja-luvatta-jyvaskylassa/1250642

The Game crafter webshop. *Elevation of Privilege.* Accessed on 7th March 2015. Retrieved from: https://www.thegamecrafter.com/games/elevation-of-privilege

Germanwings Plane Crash Investigation Press Conference. Accessed on 28 March 2015. Retrieved from http://www.theguardian.com/world/live/2015/mar/26/germanwings-plane-crash-investigation-press-conference-live-updates-4u9525

Germanwings crash: Airline cockpit doors locked since 9/11. Accessed on 28 March 2015. Retrieved from http://tribune.com.pk/story/859378/germanwings-crash-airline-cockpit-doors-locked-since-911-attacks/

Giles, L. 1910. *Sun Tzu on the Art of War*. Accessed on 28 March 2015. Retrieved from http://artofwarsuntzu.com/Art of War PDF.pdf

Harris, S. 2013. *CISSP Exam Guide*. 6th ed. USA: McGraw-Hill.

Houding, D., Casey, T., Rosenquist,M. 2012. *Improving Healthcare Risk Assessments to Maximize Security Budgets*. Accessed on 5th November 2014. Retrieved from https://www-ssl.intel.com/content/www/us/en/healthcare-it/risk-assessments-maximize-security-budgets-brief.html

Improving Web Application Security: Threats and Countermeasures - June 2003. Accessed on 23 July 2015. Retrieved from https://msdn.microsoft.com/en-us/library/ff648644.aspx

Jackson, G.M. 2012. *Predicting Malicious Behaviour: Tools and Techniques for Ensuring Global Security*. Indianapolis, USA: Wiley.

Kearney, Albert J. *Understanding Applied Behavior Analysis: An Introduction to ABA for Parents, Teachers, and Other Professionals*. London, GBR: Jessica Kingsley Publishers, 2007. ProQuest ebrary. 22 April 2015.

Kohnfelder. L, Garg, P. 1999. *The threats to our products on MSDN Blogs*. Accessed on 16 August 2015. Retrieved from https://blogs.msdn.com/cfs-filesystemfile.ashx/__key/communityserver-components-postattachments/00-09-88-74-86/The-threats-to-our-products.docx

Koistinen, P. 2013. *Security Model for Agile Software Development. Laurea University of Applied Sciences, Degree Programme on Security Management*. Accessed on 6 August 2015. Retrieved from http://urn.fi/URN:NBN:fi:amk-2013111116849

Larcom, B. 2012. *Threat Modelling Using Trike. Air Mozilla Video*. Accessed on 13 August 2015. Retrieved from https://blog.mozilla.org/security/2012/02/28/brenda-larcom-presentation-on-threat-modeling-using-trike/

McSweeney F.K., Murphy E.S. 2006. *The Wiley Blackwell Handbook of Operant and Classical Conditioning*. Chichester, UK: Wiley.

Miller, L. K. 2006. *Principles of everyday behavior analysis*. 4th ed. Belmont, CA: Thomson, Wadsworth.

Morana, M. 2014. *Process for Attack Simulation and Threat Analysis: Engineering Attack Resilient Software & Applications. Info Security Europe Conference 2014*. Accessed on 7 August 2015. Retrieved from http://www.infosecurityeurope.com/__novadocuments/87663

Narciso, I. 2013. *Michael Schumacher Health Update: Doctor's Laptop Containing His Sensitive Data Stolen, Family Fears Leak of His True Medical Condition*. Page on The Gospel Herald 11 March 2015. Accessed on 23 May 2015. Retrieved from http://www.gospelherald.com/articles/54692/20150311/michael-schumacher-update-doctor-s-laptop-containing-his-sensitive-data-stolen-family-fears-leak-of-his-true-medical-condition.htm

Manifesto for Agile software development. 2001. Accessed on 2 August 2015. Retrieved from http://agilemanifesto.org/

McAfee, J. 2015. *John McAfee: Ashley Madison database stolen by lone female who worked for Avid Life Media*. Accessed on 3 September 2015. Retrieved from http://www.ibtimes.co.uk/john-mcafee-ashley-madison-database-stolen-by-lone-female-who-worked-avid-life-media-1516833

Michael Schumacher injured in skiing accident in France. Page on BBS Sport's website 29th December 2013. Accessed on 23 May 2015. Retrieved from http://www.bbc.com/sport/0/formula1/25542340

Microsoft Security Development Lifecycle Core Training Classes: The PowerPoint material version 1 / 23th February 2010. Accessed on 29 July 2015. Retrieved from http://www.microsoft.com/en-us/download/details.aspx?id=16420

Mitre. 2014. CAPEC home page. Accessed on 16 August 2015. Retrieved from https://capec.mitre.org/

MyAppSecurity. Accessed on 22 July 2014. Retrieved from http://myappsecurity.com/comparison-threat-modeling-methodologies/

OWASP Main Page. Accessed on 26 July 2015. Retrieved from https://www.owasp.org/index.php/Main_Page

OWASP Threat Risk Modelling. Accessed on 23 July 2015. Retrieved from https://www.owasp.org/index.php/Threat_Risk_Modeling

PwC. (30.9.2014). *The Global State of Information Security Survey 2015*. Accessed on 28 March 2015. Retrieved from PwC: http://www.pwc.com/gx/en/consulting-services/information-security-survey/download.jhtml

Rosenquist, M., 2009. *Whitepaper: Prioritizing Information Security Risks with Threat Agent Risk Assessment*. Accessed on 5th November 2015. Retrieved from: https://communities.intel.com/docs/DOC-4693

Saitta, P. Larcom, B. 2012. *Trike home page*. Accessed on 13 August 2015. Retrieved from http://octotrike.org/home.shtml

Saitta, P. Larcom, B. 2012. *Trike tools*. Accessed on 13 August 2015. Retrieved from http://octotrike.org/tools.shtml

Saitta, P., Larcom, B., Eddington, M. 2005. *Trike v.1 Methodology Document [Draft]*. Accessed on 13 August 2015. Retrieved from http://octotrike.org/papers/Trike_v1_Methodology_Document-draft.pdf

Saunders, M., Lewis, P. & Thornhill, A. P. 2012. *Research Methods for Business* Students. 6th, ed. Rev. ed. Essex: Pearson.

SDL Threat Modelling Tool. Accessed on 26 July 2015. Retrieved from https://www.microsoft.com/en-us/sdl/adopt/threatmodeling.aspx

SDL Team. 2014. *Introducing Microsoft Threat Modelling Tool 2014. The page on the Microsoft blogs*. Accessed on 3 August 2015. Retrieved from http://blogs.microsoft.com/cybertrust/2014/04/15/introducing-microsoft-threat-modeling-tool-2014/

Security Development Lifecycle. Accessed on 26 July 2015. Retrieved from Microsoft http://www.microsoft.com/security/sdl/resources/evolution.aspx

Shostack, A. 2014. *Threat Modelling: Designing for Security*. Indianapolis, USA: Wiley

Shostack, A. 2010. *Elevation of Privilege: The easy way to threat model. Youtube video*. Youtube: Christiaan008. Accessed on 22 August 2015. Retrieved from https://www.youtube.com/watch?v=gZh5acJuNVg

Stubb: Russia is trying to irritate us. Yle news. Accessed on 28 April 2015. Retrieved from http://yle.fi/uutiset/stubb_russia_is_trying_to_irritate_us/7442963

Vatanen, M. 2014. *Master's Thesis - Intrusion Detection During IT Security Audits*. Accessed on 30 March 2015. Retrieved from http://urn.fi/URN:NBN:fi:amk-2014110615317

Noronen, H-P. 2014. *Improving Performance, Quality and Happiness of Software Development Team: Agile and Lean approach*. Accessed on 2 September 2015. Retrieved from http://urn.fi/URN:NBN:fi:amk-2014061012363

Velez, T.U., & Morana, M.M. 2015. *Risk Centric Threat Modelling. Process for Attack Simulation and Threat Analysis*. New Jersey: Wiley

Walker, M. 2014. *CEH Certified Ethical Hacker Exam Guide*. 2nd ed. USA: McGraw-Hill.

Walter, E. (Ed.), Woodford, K. (Ed.), Good, M. (Ed.). *Cambridge Advanced Learner's Dictionary*. 2008. 3rd ed. Cambridge, UK: Cambridge University Press.

Web Application Security Consortium (WASC). 2010. WASC Threat Classification v2.0. Accessed 5th September 2015. Retrieved from http://projects.webappsec.org/f/WASC-TC-v2_0.pdf

Xbox 'Hacker' Reveals Why He Attacked Consoles. Sky news. Accessed on 21 January 2015. Retrieved from http://news.sky.com/story/1398435/xbox-hacker-reveals-why-he-attacked-consoles

Zetter, K. 2014. *Hospital Networks Are Leaking Data, Leaving Critical Devices Vulnerable*. Accessed on 18 July 2014. Retrieved from http://www.wired.com/2014/06/hospital-networks-leaking-data/

APPENDICES

APPENDIX A. STRIDE

Public

STRIDE

The threat model by Microsoft

Aki Sivula

Test Manager
Tieto, Industrial Internet
aki.sivula@tieto.com

Public

Content

- Why?
- STRIDE history
- Acronym STRIDE
- Advantages
- Disadvantages
- Usage
- Examples
- Questions?

Length

30min – 60min

Why?

- Typically, old software is not safe (compare Windows 98 to Windows 7)
- The new software will not get better without basic security knowledge
- Interest in the health care data increases all the time
- Because you do not want to get onto Iltalehti / Expressen frontpage due to your mistake

STRIDE history

- MSDN blogs: The threats to our products, Kohnfelder & Garg. 1st April 1999.
- The model defined by The Microsoft Security Task Force (invented by Kohnfelder & Garg 1999)
- Microsoft used also other security models: at least DREAD that deceased 2010.
- Part of the SDL Threat Modeling Process started in 2002 - 2003

Acronym STRIDE stands for…

Threat	Property violation
Spoofing	Authentication
Tampering	Integrity
Repudiation	Non-repudiation
Information disclosure	Confidentiality
Denial of service	Availability
Elevation of privilege	Authorization

STRIDE threats and violations (adapted from Shostack, 2014, 62)

Advantages

- Investigation of the right defences for the possible attacks on the target system
- Possibility to use within attack libraries like CAPEC or OWASP -> may help a lot!
- No need to be a security nerd so suits for you ☺
- Microsoft Threat Modeling Tool 2016 supports STRIDE directly
- Easier to find defenses than on CAPEC (Shostack, 2014, 106).

Disadvantages

- Might not be as lightweight as it sounds.
- Not the best (=easiest) choice as first model, well structured models are easier in the beginning.
- The vulnerabilities and the management of vulnerabilities not throughout covered
- Doesn’t take automation into account -> after all, investigation of the same issues is sometimes tedious

Usage

- Select the target system
- Select diagram that contains interfaces
- For instance, select one interface of the system and use the whiteboard.
- Microsoft Threat Modeling Tool 2016 for advanced usage
- Apply spoofing, tampering, repudiation, information disclosure, denial of service or elevation of privilege to the interfaces / system

Tips for usage

- Better understanding of the target users helps to understand what the important security flaws are
- Eventually, think about the whole system and connected systems. What could the attacker do?
- Think “out of the box”, not only interfaces
- In the beginning: Elevation of Privilege card game
- Later: Threat Modeling Tool 2016

Spoofing examples

- User identity (Fake users, fake doctor etc.)
- Create file/s (executable, library or config, link)
- IP packet spoofing (man in the middle attack, ARP, IP, DNS, IP redirection)
- Poisoning attacks (untrusted updates, like Notepad++)
- Spoofing + replay attack
- E-mail (confidentiality and integrity)

STRIDE threats and violations (adapted from Shostack, 2014, 65)

Tampering examples

- Catch the packet with Firefox "Tamper data" add on, modify it and shoot (=packet injection attack)
- Tamper memory or the file (cracking)
- Edit the initilization, configuration, registry whatever without authentication (no integrity / sanitation checks)
- Data corruption
- Redirect the data flow or modify it (WiFi, 4G)

STRIDE threats and violations (adapted from Shostack, 2014, 67)

Repudiation examples

- Undetected login attempts (no logon audits)
- No log entries that the file was submitted by the authorized person?
- I did not browse my ex husband's health data …
- Who browsed on the xxx sites? Not me!
- Deletion of any sensitive data without permission
- We delivered the update batch in agreed time

STRIDE threats and violations (adapted from Shostack, 2014, 69)

Information disclosure examples

- Private data on social media
- Access to the protected data without authorization
- Sniffing the network (Wireshark), Remote Access Trojans
- Unencrypted protocol: identity, location, passwords, chat messages, email …
- Poor encryption (it is not good idea to use Enigma anymore…)
- Too informative error messages, unencypted protocol …

STRIDE threats and violations (adapted from Shostack, 2014, 70 - 71)

Denial of service examples

- Prevent the website usage with DDoS
- 100% CPU load
- Fill the hard disk with needless garbage
- Crash or error -> service doesn't start up anymore
- Fill up the email box with spam
- Download warez on the company network (network resources)

STRIDE threats and violations (adapted from Shostack, 2014, 72)

Elevation of privilege examples

- Possibility to run actions that are meant for privileged users
- Corrupt the process: "*Send inputs that the code doesn't handle properly*", (Shostack, 2014, 73)
- Missing, improper or buggy authorization checks: access to the administrator pages
- Tamper config files and gain authorization

STRIDE threats and violations (adapted from Shostack, 2014, 73)

Tips before the game

- Just collect the things that might go wrong
- Technical details can be investigated later
- It is easy to find threats but often it is difficult to put them to the ”right” category so don’t worry too much about the categories

References

- Shostack, A. 2014. *Threat Modelling: Designing for Security*. Indianapolis, USA: Wiley
- Shostack, A. 2010. *Elevation of Privilege: The easy way to threat model. YouTube video*. YouTube: Christiaan008. Accessed on 22 August 2015. Retrieved from https://www.youtube.com/watch?v=gZh5acJuNVg
- *SDL Threat Modelling Tool*. Accessed on 26 July 2015. Retrieved from https://www.microsoft.com/en-us/sdl/adopt/threatmodeling.aspx
- *Security Development Lifecycle*. Accessed on 26 July 2015. Retrieved from Microsoft http://www.microsoft.com/security/sdl/resources/evolution.aspx

Public

Changing perspectives™

Aki Sivula

Test Manager
Tieto, Industrial Internet
aki.sivula@tieto.com

APPENDIX B. CAPEC

Public

CAPEC

The attack library by MITRE

Aki Sivula

Test Manager
Tieto, Industrial Internet
aki.sivula@tieto.com

Public

Content

- What is attack library?
- CAPEC
- Advantages
- Disadvantages
- Usage
- Domains of attack, Mechanics of attacks
- Applying CAPEC
- Tips

Length

Max 15min

What is attack library?

- A library that contains variety of attacks. No strict definition.
- For instance, attack tools, attack patterns, proof-of-concept code or complete exploit code.
- Some libraries contain attack models but some contain only attacks without modeling

CAPEC

- Common Attack Pattern Enumeration and Classification
- Over 460 different attack patterns
- Categorized by mechanisms of attack or domains of attack
- Supports Google search
- Sponsored by U.S. Department of Homeland Security
- Classification of common attacks

Advantages

- Attacks are represented as patterns that are applicable in different contexts
- Each entry completion and status is shown on the attack.
- One of most comprehensive attack libraries available
- Is not limited to web applications only
- Continuously improved and maintained
- Customized attack library possible

Disadvantages

- May lead in less creative thinking
- CAPEC might feel boring and/or difficult in beginning
- CAPEC offers classification of common attacks, STRIDE offers security properties (Shostack, 2014, 106).
- It is easier to recognize defences with STRIDE than with CAPEC

Usage 1/2

- Many different ways to use, team should select and agree appropriate way. One way is to use CAPEC's two views.
- Organized by mechanisms of attack http://capec.mitre.org/data/definitions/1000.html
- Organized by attack domains http://capec.mitre.org/data/definitions/3000.html

Usage 2/2

- Direct usage:
  In clear cases just select the:
  CAPEC-66 SQL Injection, CAPEC-103: Clickjacking,
  CAPEC-62: Cross Site Request Forgery …
- Possibility to derive attack patterns to the own library (might consume vast amount of time).

Domains of attack

CAPEC VIEW: Domains of Attack

View ID: 3000
Structure: Graph

View Objective

This view organizes attack patterns hierarchically based on the attack domain.

Relationships

3000 - Domains of Attack

- Social Engineering - *(403)*
- Supply Chain - *(437)*
- Communications - *(512)*
- Software - *(513)*
- Physical Security - *(514)*
- Hardware - *(515)*

Mechanisms of Attack

CAPEC VIEW: Mechanisms of Attack

View ID: 1000
Structure: Graph

View Objective

This view organizes attack patterns hierarchically based on mechanisms that are frequently employed when exploiting a vulnerability.

Relationships

Expand All | Collapse All

1000 - Mechanisms of Attack

- Gather Information - *(118)*
- Deplete Resources - *(119)*
- Injection - *(152)*
- Deceptive Interactions - *(156)*
- Manipulate Timing and State - *(172)*
- Abuse of Functionality - *(210)*
- Probabilistic Techniques - *(223)*
- Exploitation of Authentication - *(225)*
- Exploitation of Authorization - *(232)*
- Manipulate Data Structures - *(255)*
- Manipulate Resources - *(262)*
- Analyze Target - *(281)*
- Gain Physical Access - *(436)*
- Malicious Code Execution - *(525)*
- Alter System Components - *(526)*
- Manipulate System Users - *(527)*

Applying CAPEC

- Easier when you have a clue what to search
- System diagram or very good knowledge of system might be benefit
- Apply existing STRIDE findings for the CAPEC
- Apply OWASP TOP-10 attacks towards your system but with the CAPEC attack patterns

Tips for usage

- Start with STRIDE and use CAPEC as supportive element
- Gain different aspects with CAPEC patterns
- Try to think what typical cyber threats for your application are
- For instance: if web logs are important in the system check
  CAPEC-81: Web Logs Tampering
- Browse the database and see what it has to offer to you

References

- Shostack, A. 2014. *Threat Modelling: Designing for Security*. Indianapolis, USA: Wiley (104 – 107).
- Velez, T.U., & Morana, M.M. 2015. *Risk Centric Threat Modelling. Process for Attack Simulation and Threat Analysis*. New Jersey: Wiley (462 - 467)

Public

Changing perspectives™

Aki Sivula

Test Manager
Tieto, Industrial Internet
aki.sivula@tieto.com

APPENDIX C. Questionnaire: Security risk and threat models

Questionnaire: Security Risk and Threat Models

This questionnaire aims to collect experiences of the selected threat models for agile development. The models were selected based on literature analysis and the original requirements were given by Tieto's representative. The data in the questionnaire will be used for the Master's thesis "Security Risk and Threat Models for Health Care Product Development Processes" for JAMK University of Applied Sciences.

*Required

Background data

---

1. Role *

   Mark only one oval.

   - ◯ Developer
   - ◯ Test engineer
   - ◯ Scrum master
   - ◯ Product owner
   - ◯ Project manager
   - ◯ Ot- ..... her:

2. Did you participate to the threat model presentation held for your team? * Mark only one oval.

   - ◯ Yes
   - ◯ No
   - ◯ Partially

3. Does the used material support understanding of the selected threat models? *

   STRIDE & CAPEC PowerPoint presentations, The Elevation of Privilege card game
   Mark only one oval.

	1	2	3	4	5	
Poor	◯	◯	◯	◯	◯	Excellent

4. Do you have a cybersecurity background? *

   For instance: cybersecurity training, hobbyist or work experience. Mark only one oval.

	1	2	3	4	5	
No security related background	◯	◯	◯	◯	◯	Security area professional

STRIDE

The applied threat model STRIDE ...

5. is easy to understand and apply? * Mark only one oval.

	1	2	3	4	5	
Easy	◯	◯	◯	◯	◯	Hard

6. supported me to recognize the risks for the product? *
   Mark only one oval.

	1	2	3	4	5	
Poor	◯	◯	◯	◯	◯	Excellent

7. supported me to recognize the threats for the product?
   * Mark only one oval.

	1	2	3	4	5	
Poor	◯	◯	◯	◯	◯	Excellent

8. supported me to recognize the vulnerabilities for the product? * Mark only one oval.

	1	2	3	4	5	
Poor	◯	◯	◯	◯	◯	Excellent

9. supported me to recognize the weaknesses for the product? * Mark only one oval.

	1	2	3	4	5	
Poor	◯	◯	◯	◯	◯	Excellent

10. suits for agile development? * Mark only one oval.

	1	2	3	4	5	
Poor	◯	◯	◯	◯	◯	Excellent

CAPEC

The applied threat model CAPEC ...

11. is easy to understand and apply? *

Mark only one oval.

	1	2	3	4	5	
Easy	◯	◯	◯	◯	◯	Hard

12. supported me to recognize the risks for the product? * Mark only one oval.

	1	2	3	4	5	
Poor	◯	◯	◯	◯	◯	Excellent

13. supported me to recognize the threats for the product? * Mark only one oval.

	1	2	3	4	5	
Poor	◯	◯	◯	◯	◯	Excellent

14. supported me to recognize the vulnerabilities for the product? * Mark only one oval.

	1	2	3	4	5	
Poor	◯	◯	◯	◯	◯	Excellent

15. supported me to recognize the weaknesses for the product? * Mark only one oval.

	1	2	3	4	5	
Poor	◯	◯	◯	◯	◯	Excellent

16. suits for agile development? * Mark only one oval.

	1	2	3	4	5	
Poor	◯	◯	◯	◯	◯	Excellent

Advantages, disadvantages and improvement proposals

17 STRIDE threat model advantages? *

.....

.....

.....

.....

.....

18. STRIDE threat model disadvantages? *

19. CAPEC threat model advantages? *

20. CAPEC threat model disadvantages? *

21. Improvement proposals regarding to the threat models usage?

For instance: Would you like to use some other way during sprints like threat modelling software, planning on the whiteboard or something else?

22. Improvement proposals regarding to the training?

.....

.....

.....

.....

.....