Analyzing Security Decisions with Discrete Event Simulation

Magnus Felde

Masteroppgave
Master i informasjonssikkerhet
30 ECTS
Avdeling for informatikk og medieteknikk
Høgskolen i Gjøvik, 2010

Avdeling for
informatikk og medieteknikk
Høgskolen i Gjøvik
Postboks 191
2802 Gjøvik

Department of Computer Science
and Media Technology
Gjøvik University College
Box 191
N-2802 Gjøvik
Norway

Analyzing Security Decisions with Discrete Event Simulation

Magnus Felde

28th June 2010

Abstract

As organizations become increasingly more dependent on information security in order to succeed, the security decisions made by the Chief Information Security Officer (CISO) also becomes important and needs to be considered in the context of the organization. However, since the complexity of the organization's internal processes and the threats the organization is facing, the CISO needs a decision making tool or method in order to determine the effects of a specific security decision. Because of this, we have in this thesis determined the suitability of utilizing Key Performance Indicators (KPIs) and Discrete Event Simulation (DES) as a method to help the CISO make the "best" security decision for his organization.

The thesis is based on a health care specific scenario which has been constructed in collaboration with Akershus University Hospital (Ahus), Rheumatism Hospital at Lillehammer and Buypass. The scenario includes a patient treatment process and the processes related to the usage of smart cards and passwords as authentication mechanisms. Furthermore, KPIs which focuses on *time usage* and *number of deviations* has been identified, where deviations within this health care specific scenario relates to more traditional security incidents.

A case study was then conducted based on the scenario. The results of this case study indicate no statistical significant difference between the two authentication mechanisms with regards to the average time a doctor uses on a business activity. However, based on the number of deviations identified, smart cards were determined the preferred security measure of the two.

In order to determine the suitability of the simulation approach, a second case study was also conducted. This second case study was based on the same scenario, but this time with a non-simulation approach. By comparing the process surrounding the two case studies, the non-simulation approach were determined the most cost-effective approach and the approach which provided the most direct link between the input data and the results. Based on this, the non-simulation approach was also determined the most suitable approach. However, we did determine that for "what if" analysis, the simulation approach becomes the best choice of the two.

Should a "what if" analysis be desirable, we have in this thesis proposed a new methodology which modelers can utilize in order to reduces the complexity of the model building process. The methodology, called Minimalistic Model Design (MIMD), excludes the temporal relationship between the identified business activities within the business process. This exclusion helps to reduce total time used on the model building process, and enables better scalability.

Sammendrag

Ettersom organisasjoner blir stadig mer avhengige av informasjonssikkerhet for å kunne lykkes er det viktig at arbeidet som gjøres innen informasjonssikkerhet tilpasses organisasjonen. På grunn av den nødvendige tilpassingen må også sikkerhetsbeslutninger som gjøres av sikkerhetslederen vurderes i en organisasjonssammenheng. Kompleksiteten i organisasjonens og de trusler en organisasjonen står overfor gjør at sikkerhetslederen trenger verktøy og metoder for å fastslå hvilken effekt en avgjørelse har på organisasjonen. Vi har på grunn av dette valgt å se på nytteverdien av å benytte Key Performance Indicators (KPIer) og Discrete Event Simulation (DES) som en metode for å hjelpe sikkerhetslederen i å gjøre de "beste" sikkerhetsbeslutningene for sin organisasjon.

Avhandlingen er basert på et scenario fra helsevesenet. Scenarioet er konstruert i samarbeid med Akershus universitetssykehus (Ahus), Revmatisme sykehuset på Lillehammer og Buypass. Scenarioet inkluderer en pasientbehandlingsprosess og prosessene knyttet til bruk av smartkort og passord som autentiseringsmekanismer. Videre er KPIer som inkluderer tid og antall avvik identifisert. I dette scenarioet er avvikene relatert til tradisjonelle sikkerhetshendelser.

En tilfellestudie ble så utført basert på scenario som ble konstruert. Resultatene av denne undersøkelsen indikerer ingen statistisk signifikant forskjell mellom de to autentiseringsmekanismer i forhold til den gjennomsnittlige tiden en lege bruker på en arbeidsoppgave. Derimot viser resultatene at smartkort er den foretrukne sikkerhetsmekanismen basert på et færre antall inntreffende avvik.

For å avgjøre hvor egnet simulering er for å avgjøre hvordan en sikkerhetsavgjørelse påvirker organisasjonen, ble en ny tilfellestudie gjennomført. Tilfellestudie er basert på det samme scenarioet, men denne gangen ble en tilnærming basert på en analytisk fremgangsmåte tatt. Ved å sammenligne prosessen rundt de to tilfellestudiene ble den analytiske tilnærming ansett som den mest kostnadseffektive av de to tilnærmingene. Den analytiske tilnærmingen ga i tillegg den mest direkte koblingen mellom input dataene og resultatene. Disse resultatene gjorde videre at vi kunne fastslå at den analytiske tilnærmingen er den tilnærmingen som er mest egnet til å avgjøre effekten av en sikkerhetsavgjørelse. Allikevel ser vi at for såkalte "What if"-analyser, så er simulering den tilnærmingen som det beste valget.

Skulle det være ønskelig å lage en simuleringmodell har vi i denne avhandlingen foreslått en ny metode som kan brukes i forbindelse med forberedelser til selve byggingen av modellen. Metoden, som heter Minimalistic Model Design (MIMD), forenkler byggeprosessen, og selve modellen, ved å utelukke det tidsmessige forholdet som eksisterer mellom ulike aktiviteter i en forretningsprosess. Ved å ekskludere dette forholdet kan vi redusere den totale tiden man bruker på modellbyggeprosessen samt at metoden bidrar til en modell som håndterer skalerbarhet bedre.

Acknowledgments

During the period of writing this master thesis many people have been involved, and without them, the work which has been done had not been possible.

Great thanks goes to my supervisor, Einar Snekkenes, who provided the initial problem description and who have been a great resource during the whole master thesis process with regards to the report itself, the methodology chosen and the simulation runs conducted. My co-supervisor, Nils Kalstad Svendsen, deserves thanks for his contributions throughout the master thesis period.

A great thanks also goes to Tina Steen who, due to her patience during these lasts months, have made it possible to work at all hours of the day. She has also helped me focus on other elements during this period, and for this I am grateful.

I would like to thank Buypass, and in specific Sverre Sandernes, who have provided me with his insight and contributions during the whole process and guided me in the direction of the health sector. Furthermore, I would like to thank Rolf Kulstad at the Rheumatism Hospital, Halvor Sandodden, Ellef Mørk, Fredrik Dahl and Lene Berge Holm at Akershus University Hospital who have all provided me with great insight. Their contribution have helped me tremendously and provided a solid foundation for the thesis. I would also like to thank Beate M. Huseby with the Norwegian directory of Health for her contributions and Fridtjov Tjemsland at Ergo Group for helping me gaining important input data. Thanks also to *Imagine That Inc.* who provided me with a full version of their simulation tool *ExtendSim* free of charge during my master thesis.

A thanks also goes to my student opponent, Morten Bye, for his feedback on my thesis, and to Kirsi Helkala for providing me with insight into her work. My co-students also deserve thanks for helped me during troubling stages of the process and for making the long days at the master lab more interesting. Finally, I would also like to thank Tone Hoddø Bakås at NorSIS and Jan Erik Østvang at SecCon for taking the time to discuss potential collaborating partners in the early stages of the master thesis process.

To those already mentioned and everyone else who have contributed during my master thesis, this master thesis could not have been done without your help and for this I am forever grateful.

Magnus Felde, 28th June 2010

Contents

List of Figures

List of Tables

1 Introduction

In this Chapter we will give a introduction to the topic covered by the thesis before we describe the problem we are to solve. The motivations for the work and the identified research questions are then presented. Finally in this Chapter, we present a summary of our contributions.

The remainder of this thesis is structured as follows.

Chapter 2	Gives a introduction to the areas of Key Performance Indicators, Business Process Modeling and Simulation, and the related work identified within these areas.
Chapter 3	Presents our choice of method based on the related work and the task at hand.
Chapter 4	Presents the modeling and simulation methodology used in this thesis.
Chapter 5	Present the selected simulation environment and how the selection process was conducted.
Chapter 6	Presents a hospital specific scenario which have been created.
Chapter 7	Results from conducting a case study based on the scenario and the methodology is presented.
Chapter 8	Describes a new methodology for model design and data collection developed based on the experiences of the case study.
Chapter 9	Compares the simulation approach with a non-simulation approach to determine the suitability of simulation.
Chapter 10	Lists the contributions made in this thesis.
Chapter 11	Provides a discussion based on the findings.
Chapter 12	Contains a list of suggested future work.
Chapter 13	Presents our conclusion of this thesis

1.1 Topic covered by the thesis

We will in this thesis determine the suitability of simulation as an approach for analyzing the effects a security decision has on an organization. In order to determine this, we will combine the use of Key Performance Indicators (KPIs), Business Process Modeling (BPM) and simulation. The KPIs are used to measure the effects of the security decision, while BPM and simulation is applied in order to understand how the organization functions, and to be able to represent the system and imitate the real-world processes.

Specifically, we will conduct a Discrete Event Simulation (DES) based on a scenario which focuses on how the two authentication mechanisms passwords and smart cards affect the patient treatment process of a Norwegian hospital. By utilizing Common Random Numbers (CRN) and the method of batch means, we will compare the confidence interval of the identified measures of performance in order to determine which of the two authentication mechanisms are "best". Depending on what is desirable for the specific KPI, "best" referrers to the security measure which increases or decreases the value of the identified KPIs the most. It is important to notice that we in this thesis do not intend to conduct an actual assessment of the implementation of the two security measures, and hence we are able to make some simplifications throughout the thesis

without affecting our objectives.

1.2 Keywords

Information security, security management, business management, simulation and modeling, business process modeling, key performance indicators

1.3 Problem description

Traditionally, when a security decision is made, the decisions implications with regards to the overall security level is consider. That is, which risks are mitigated as a result of the decision, and which residual risks are considered acceptable. However, a security decision must also be made with consideration of the context of the business operations and day-to-day activities. If such considerations are not made, a security decision could reduce the effectiveness of daily activities, and in a worst case scenario, prohibit the organization in conducting its main activities.

However, the complexity of the interacting processes found within an organization makes it difficult to anticipate the effects security decisions has on a organization. Because of the complexity, the Chief Information Security Officer (CISO) or others who are responsible for making security decision needs a decision making tool in order to assist them in the process. Several different approaches can be made when determining how a decision affects the organization, where modeling and simulation is one such approach. Such an approach is much used in process performance and optimization analysis, e.g. Jacobson et al. ([39]) and Holm et al. ([29]). However, whether this approach is suitable for the task of determining the effects of security decisions on a organization is unclear.

1.4 Justification, motivation and benefits

The purpose of information security, and information technology (IT) in general, is supporting the business processes in such a way that the organization achieves its goals. Therefore, in addition to ensure that a organization meets its confidentiality, availability, and integrity objectives, Jaquith ([40]) states that information security, and security controls in particular, must be considered in the context of the business in which they operate. That is, the objective of the CISO is to provide value to top management and shareholders, and the CISO therefore needs to understand the goals of the organization and how the security decisions may influence these goals.

In order to achieve this, one must understand how the business processes functions and how the security decisions affect these processes. According to Parmenter et al. ([63]), we are able to determine this affect by utilizing measures such as KPIs which incorporates those aspects of organizational performance that are the most critical for the current and future success of the organization.

However, the complexity found within an organization makes determining the cause-and-effect of security decisions difficult. In order to determine the effects of controls currently implemented, or those which is considered to be implemented, without disrupting the business processes, we need a tool or a method which allows us to make a representation of the system and imitate the real-world processes. Simulation then becomes a candidate for achieving such goals, since we gain a insight into how different configurations affect the business, without

actually disrupting the business.

The demand for automatic tools, commonly accepted metrics and to better understanding of how the business is affected by security problems and controls are not new issues in the field of information security, see e.g. Butler et al. ([8]), Cohen ([12]) and Neubauer et al. ([49])). However, these are issues which needs to be combined and resolved.

1.5 Research questions

We will in the thesis attempt to answer the following three identified research questions:

1. To what extent does information security management decisions influence the organizational goals?
2. To what extent are KPIs suitable for measuring the affects of security decisions?
3. To what extent is the construction and evaluation of a simulation model a suitable approach in determining how security decisions affect the goals of an organization?

1.6 Boundaries of the thesis

Since the topic of simulation and decision making is quite broad, a boundary is required. Therefore, in order to answer our research questions we will only consider the decision of which authentication mechanism that should be implemented. In specific, we will determine whether or not to implement smart cards as an authentication mechanism, as opposed to using passwords. Furthermore, this decision will be made based on a simplified health care scenario where a patient treatment process will be considered. We will make the simplifying assumption that these boundaries do not affect our decision with regards to determining the suitability of the simulation approach.

1.7 Summary of contributions

In this thesis we have defined a health care specific scenario and conducted a case study which determined the effects of implementing smart cards compared to using passwords in the scenario. Based on the experiences we gained during this case study, we have also developed a new methodology for model design and data collection which reduces the overall complexity. A second case study was conducted based on the same hospital specific scenario, but this time with a non-simulation approach. Based on the two case studies, we have further been able to compare the suitability of both the simulation approach and the non-simulation approach. A final contribution is the data set which has been collected during our scenario construction. The data set includes figures related to the duration of health care specific activities, the frequency of authentication failures with regards to smart cards and passwords. Further more, the costs of such failures with regards to the added time a employee uses on the process is included.

2 Related Work

In this Chapter we present the background material and related work in the areas applied in our thesis. The purpose of the Chapter is to give the readers an introduction into the areas which we apply, which hence allow the reader to better understand our work, and how this relates to the work of others. In specific, we will in this Chapter cover the areas of

- Key Performance Indicators (KPIs)
- Business Process Modeling (BPM)
- Control effectiveness and cost
- Simulation and modeling

For our thesis, KPIs becomes relevant as we then are able to include a measure which incorporate organizational performance, and which we can utilize in order to measure the effects of a security decision. Furthermore, BPM is relevant since we then are able to analyze the business processes and ultimately enable us to model these processes. In order to determine if a particular decision is considered "best", we also need to determine its overall affect on the organization. Because of this, the area of control effectiveness and cost becomes relevant for our thesis. Finally, since this thesis involves simulation, a natural area of interest is that of simulation. We will determine which different simulation approaches exist such that we are able to determine which of the approaches is most suitable for our task.

Although much literature is found on each issue separately and some literature combines parts of these issues, very little literature combines all of these issues. Because of this, we have chosen to structure the Chapter based on the relevant areas, and present the related work within each area.

2.1 Key Performance Indicators

Key Performance Indicators (KPIs), represent a set of measures focusing on those aspects of organizational performance that are the most critical for the current and future success of the organization [63]. According to Boynton et al. ([7]) Critical Success Factors (CSFs) are those few things that must go well to ensure success for an organization and must be given special and continual attention. In such, a good KPI will be affected by most of the core CSFs and should tell you what action need to take place. KPIs can be both financial and non-financial measures, and they are tied to an organization's strategy by typically using concepts or techniques such as the Balanced Scorecard (BSC)[1] [80]. According to Parmenter ([63]), a good KPI should in fact affect more than one BSC perspective[2].

[1]The goal of BSC is to align business activities to the vision and strategy of the organization and to monitor organization performance against strategic goals [37]

[2]The four perspectives are *Financial*, *Customer*, *Learning and Growth* and *Business Processes*.

The Cobit framework [20] states that KPIs, described as Performance indicators, indicate whether goals are likely to be met. Furthermore, they can be measured before the outcome is clear and, therefore, are called "lead indicators". The fact that they can be measured before the outcome is clear is important since this allows us to determine the effect of security decisions in advance. Further, it is stated in [20] that KPIs are measurable indicators of performance of the enabling factors of IT processes, and indicates how well the process enables the goal to be reached. We can hence analyses how the security controls and threats affect the process, and hence also the goal of the organization.

As security is a process, and since processes are measured by metrics and key indicators, we need to think about security in the same way that is done in other types of disciplines, namely as activities that can be named, and whose efficiencies can be measured with key indicators [40].

These key indicators should incorporate time and money measures, should be measured consistently, and should be comparable across companies to facilitate benchmarking [40]. Furthermore, we have to measure not only the incidents and controls people apply, but also the threats if we want to know why some incidents happen and others don't [40].

One of the motivators for focusing on the KPIs of an organization is that it is widely agreed that it is better with a few good indicators rather than many poor indicators [21]. The KPIs used in a organization are often well thought-through and are hence often good indicators. Another motivation for using KPIs are that since these are used in most organizations already, and understood by the management, they allow the security manager to communicate better with the rest of the management group. They get a common language and it is easier to determine the true effect of the security measures on the organization.

In the paper [5], Bartolini et al. utilize KPIs in a decision support tool which conducts a impact analyses on the business based on the actions performed in the IT systems and processes. A information model for defining business objectives and the KPIs which the objectives are based upon is provided. As stated in [5], the enterprise needs to drive incident prioritizing from its business objectives and evaluate the impact on a business level, as well as its urgency in terms of the cost to the business of not dealing with it in a timely fashion. In order to do this, a Management by Business Objectives (MBO) information model has been developed.

The MBO information model is articulated around a set of key concepts, namely Objectives, KPI and Perspectives. Objectives correspond to the Cobit's Key Goal Indicators (KGIs) and express one or more target values over a KPI. Perspectives bundle objectives together that concern a certain angle of the business, e.g. financial perspective or customer perspective.

It is stated in [5] that one can either use alignment with the objectives as a measure of utility to rank alternative management options, or one can alternatively use a monetization process which is useful in that it allows instant comparison with measures of the monetary cost of executing the option. In addition, one needs to identify "episodes" that can have an impact on the KPIs, where episodes are described in terms of the metrics underlying the KPI.

Once business impact of the incident has been computed one is faced with the problem of prioritizing them so as to minimize the total impact on the business [5]. By using a definition of a set of priority levels that are used to classify the incident (defined by the Information Technology Infrastructure Library (ITIL) [38]) and require the user to express constraints on what are

the acceptable distributions of incidents into priority levels, the result is incident prioritizing to maximize alignment with business objectives.

A self-optimization solution based on high-level business objectives such as maximizing revenues is proposed by Aiber et al. in [1]. The optimization requires a model of the system, which is composed of three main sub-models, namely a business level model, IT model and IT-to-business level impact analysis model. The business level model supports the calculation of the business metrics and should present a single quantity that can be used to quantify the alignment of the IT with the business objectives. The IT model is composed of the system model, which covers the hardware configuration of the IT, and the system user behavior model, which takes into account the manner in which the users of the IT infrastructure use the systems supported by this infrastructure. Finally, the IT to business level impact analysis model defines how events at the IT level impact the business objectives defined by the business level model.

Statistical methods are used in order to detect abnormal situations, e.g. failure of a server. Several key business metrics are chosen to be tested for an abnormal situation, where the metrics in the actual environment are constantly compared with the results for these metrics in the simulated environment.

The work presented in [1] results in a clear connection of IT related policy decisions to business level metrics such as profit or ROI.

2.2 Business Process Modeling

In business, a process is a way to achieve a specific objective that is related to creating value for the end-customer, while business process modeling is the art of describing how work gets done in a company at the appropriate level to achieve the desired communication [67]. More comprehensive, a business process can be described as a network of connected activities and buffers with well-defined boundaries and precedence relationships, which utilize resources to transform inputs into outputs for the purpose of satisfying customer requirements [43].

The process comes off the drawing board and comes to life, first as a business process model and then as some form of technology that assists with implementation [67].

Visual business models are descriptions of the steps that take place during a process and are frequently represented in flowcharts [67]. The process architecture or process structure can be characterized in terms of five main components or elements according to Laguna et al. ([43]). The five main components are *Inputs and Outputs, Flow units, Network of activities and buffers, Resources* and *Information structure*. Visual modeling languages used to represent business processes include, but is not limited to, Business Process Modeling Notation (BPMN), the Unified Modeling Language (UML) and Integration Definition for Function Modeling 0 (IDEF0), Business Process Execution Language (BPEL) and Web Services Choreography Description Language (WS-CDL).

A set of process modeling success factors and measures have been identified by Bandara et al. ([3]), where user participation is the factor that are identified as the most crucial.

Recall from Section 2.1 that the papers [1] and [5] both determined the effects of their analysis of the business processes. The IT-to-business level impact analysis model in [1] enabled them to create a interface between two different levels of detail, and made it possible to align IT

with the business processes.

Furthermore, the approach described by Neubauer et al. ([49]) enables an integration of the corporate business processes that should be protected, security frameworks that allow the definition of security levels and IT-processes, and methods for the valuation of security. We will describe this paper in more details in the next Section.

2.3 Control effectiveness and cost

Security controls don't exist in a vacuum; they must be considered in the context of the business in which they operate [40]. Security controls are designed to ensure that an organization meets its confidentiality, availability, and integrity objectives. Therefore, when we speak about security effectiveness we are really talking about the effectiveness of the controls. Indicators and metrics enable us to measure this effectiveness and therefore serve as the underpinnings of a system for ensuring accountability. As stated in [40], we need to ask two questions: what hypothesis can be formed about the efficiency or effectiveness of security controls, and what evidence can be marshaled to support or disprove that hypothesis?

Olsen ([61]) argues that as security controls are countermeasures against some sort of adversary, and as such are based on sets of assumptions made with regards to the adversary by a system's designers, adversary modeling becomes of importance. "Adversary model" is the set of assumptions, explicit and implicit, which have been made with regards to the adversary in any given situation [61]. Furthermore, taking into consideration the adversaries' intent and plans are important according to Kott et al. ([41]) in order to create a strategy which is to counter the adversary.

A novel framework to quickly and efficiently get an overview over which assumptions the designers of a system have made with regards to its adversaries is introduced in [61]. The framework can be used to simplify the work of documenting and clarifying assumptions prior to and during security effectiveness analysis.

The framework, and adversary modeling in general, requires knowledge about the system in order to identify the paths of communication and current security countermeasure implementations. The modeling of systems usually consists of three distinct phases [61]. First, one has to determine principals (i.e. any individual or system/machine with which the adversary may interact in any way) and the channels (i.e. what facilitate information flow between principals) connecting them. This requires one to understand the general data flow of the system which we are able to gain through Business Process Modeling approach. The next phase is to identify the existing adversary model based on assumptions made with regards to the adversary. Key information is to find what kind of security measures are put in place in the system, and what kind of adversaries they thwart. The final phase is to identify adversaries not protected against. Key here is looking at the unprotected channels identified in step 1 and determining what operations, if any, it is feasible that an adversary may be able to execute against them.

Game Theory is a field which is highly relevant in order to understand the adversary and anticipate their next move according to Kott et al. ([41]). However, when the complexity of the environment increases, other methods than Game Theory might be more relevant for the task of creating a appropriate strategy. For instance, the two multi-agent learning algorithms which

is presented in [32], can, in combination with for example Intrusion Detection Systems (IDS), provide an additional level of security.

In [8], a structured cost-benefit process to evaluate alternative security decisions is presented. The process is based on risk assessment and utilizes elements of Utility Theory (see [82]). Furthermore, the process is designed to help mitigate faulty/bias ranking of the threats and effectiveness of the controls. It looks at different threats and controls, and which of the controls that is most effective against the threats. However, the cost-benefit process described in [8] does not consider how the controls and threats influence the business processes.

A framework for the valuation of security measures based on the external value of core business processes is developed by Neubauer et al. ([49]). Several models and frameworks for the implementation and valuation of security exists, where one can differentiate between Security Frameworks (aims at optimizing the effort needed to introduce security), Maturity Models (provides methods for the assessment and definition of security levels) and Valuation Models (focus on the valuation of security measure cost) [49]. However, these have in common that they do not consider the external business value of reaching a defined security level [49].

The approach in [49] allows an integration of corporate business processes that should be protected, security frameworks that allow the definition of security levels and IT-processes, and methods for the valuation of security. Different kinds of security costs are considered, namely investment costs, operating costs and recovery costs. Furthermore, the lost business value is also considered, and is measured based on core business processes that are affected. By using company-specific business processes, one can more accurately collect data that is needed for valuation of security cost-benefit [49].

Wei et al. ([79]) estimates the business value of a given asset under attack by focusing on the Security Management processes found within ITIL [38]. A cost-benefit analysis methodology is proposed, and a cost model based on an investigation of the cost factors and categories of various intrusions is build. By estimating the business value of a given asset under attack a decision can be made as to what to do next. The cost model, which is used in a real-time network IDS, calculates the total costs of detecting and responding to an intrusion. The total cost is an important consideration since the cost of detection and countermeasures could be much higher than the benefits, which again suggest that one should not respond to the intrusion.

By analyzing the risk for a network system, the cost model presented in [79] computes the Annual Loss Expectancy (ALE) for management and controls, and performs a cost-benefit analysis. For the assets, the analysis also determines the criticality and sensitivity, i.e. the degree of dependency, and the system's importance and vulnerability, respectively.

Torres et al. ([76]) attempts to create a universally accepted information security framework in order to measure the effectiveness of implemented security controls. The framework consists of twelve critical success factors[3], which combined consists of 76 indicators, designed for implementing and ensuring effective information security management. It is argued in [76] that the absence of such a framework prevents organizations from identifying the real mechanisms that control information security behavior.

[3]The factors presented in the paper do not align with the definition of CSFs presented in this thesis, and the factors should therefore not be mistaken with the CSFs described earlier in this Section.

Three different types of controls are identified in [76], namely technical, formal and informal controls. The effectiveness of these controls is measures, amongst others, within the CSF of *Dynamic Evaluation of Information Security Effectiveness*. However, although the framework has several CSFs, e.g. *Business Connections* and *Information Security Integration*, which aims at aligning security strategies and security controls with business goals and objectives, the framework does not determine how the security controls directly affects the organization. Because of this, the work presented in [76] needs to be directly linked together with the rest of the organization, e.g. through the organizations KPIs.

Gordon et al. ([23]) presents an economic model which determines the optimal amount of investment needed in order to protect a given set of information. The optimal amount of investment is based on the vulnerability of the information to a security breach and the potential loss should such a breach occur. It is argued that the key in analyzing information security decisions is not the vulnerability (or the expected loss without the investment), but the reduction in expected loss with the investment. By investing in information security, it is assumed that an organization can only influence the vulnerability of an information set, and not reducing the threat. The value of the information set is measured by the potential loss associated with the information set.

It is further argued in [23] that little or no information security is economically justified for extremely high, or extremely low, levels of vulnerability, and that an organization might be better off concentrating its efforts on information sets with midrange vulnerabilities. The analysis further more suggests that an organization should only spend a small fraction of the expected loss due to a security breach. More precise, the analysis performed in [23] suggest that a risk-neutral organization should never spend above 37% of the expected loss, although it is further determined that the optimal expenditures for protecting a given information set does not always increase with the increases in the information set's vulnerability.

There are a couple of simplifications in the economic model which needs to be enlightened. First of all, there is made an assumption that the incremental fixed costs of information security investment is equal to zero, which as is stated in [23], clearly played a crucial role in the analysis. Furthermore, the analysis does not consider the game theoretic aspects of information security. Finally, it is important to notice that the model presented in [23] is not intended to cover protection of assets or other circumstances where a loss could be catastrophic. Hence, the amount of investment suggested by the analysis (<37%) should in such cases not be taken into consideration. For example, within the health sector, an diviation (i.e. incident) that will not be covered by the economic model due to its consequence is *full unauthorized access to all medical records* [57].

O'Gorman ([60]) compares three different authentication methods, namely passwords, security tokens and biometrics, with regards to their effectiveness against different types of attacks and how suitable each of the authenticators are based on security specifications. Enabling cross-category comparison is important in order to determine how different methods prevent different types of attacks, and hence, determining which method is the "best". A limitation of the method described in [60], is that authentication is narrowly focused on remote computer authentication only, and does not include stand-alone PC or human gatekeepers.

A set of potential attacks are identified in the paper, and the authenticators are compared

based on whether a system is determined as strong or weak with regards to the attacks. A strong system is defined as a system where the cost of attack is greater than the potential gain to the attacker. The cost in this case includes money, time used and the potential for criminal punishment, amongst other elements. It is emphasized that while a authenticator might be strong against some types of attacks, it might be weak with regards to other types of attacks, and because of this, one needs to identify authenticator combinations that complement strengths and reduce weaknesses against different attacks.

In [60], administrative costs, compared to per-user cost and infrastructure costs, are determined as the most important cost to consider. It is stated that a convenient authenticator reduces the administrative costs, which then should be an argument for selecting an authenticator which the users finds convenient. Furthermore, the current computing infrastructure will influence the selection of authenticator as this might reduce the possible authenticators to be applied.

A method for ranking authentication products is presented in Helkala et al. ([27]). This method, which focuses on the authentication of people, builds amongst others upon the work of O'Gorman ([60]) which we described above. The work of [60] is extended by including usage scenarios which increases the applicability of the ranking [27]. Another distinction between the two papers is that in [27], authentication *products* are ranked, as opposed to authentication *methods* in [60].

When computing the security level of a product as is done by Helkala et al. ([27]), one is able to include applications that are to be used in secure, closed networks. Physical or electronic controls used to increase the barriers for the attacks can also be include in the computation. Furthermore, when considering the security level of a product, the probability of social engineering and situations where adversaries finds lost tokens can be included. Such considerations differentiate the work of [27] from others who map numerical entropy levels to security levels, e.g. [59]. Furthermore, the work of [27] does not include the limitations which were identified in [60].

The ranking of authentication alternatives is carried out by defining a distance metric, i.e. product x is n units "better than" product y. The methodology utilizes the cross-category comparison method described in [60], where the main comparison factors are security, convenience and cost. In addition, usage scenarios are, as earlier mentioned, also included. By utilizing these issues, organizations can become actively engage in the selection of authentication products, opposed to simply accepting what is offered by the vendor.

The method presented in [27] evaluates a particular product based on four stages, namely *User and environment compatibility, Security level compatibility, Usability*, and *Costs*. A product which does not fulfill a stage will be excluded for further comparison. The sequence in which the stages are performed does not affect the output, but it is argued that this particular sequence is the most cost-effective as one can exclude products early on in the method by applying calculations and processing which requires less amount of work, compared to later steps such as the cost calculations. Within each of the four stages, several parameters are identified and utilized to rank the authentication products. Such parameters include the success rate (%) of attacks, the search space of the product, how often a authenticator must be renewed or reset, the probability of human errors which increases the total time used on a authentication process, the cost of equipment used, and the enrollment costs involved, to mention a few of the many parameters

used.

2.4 Simulation and modeling

In this section we will first look closer at the process of building models, before we describe the area of simulation and the models which they are based on.

2.4.1 Model building

A model can be defined as a representation of a system for the purpose of studying the system [4].

Building a model is an iterative process and each step in the process will require comparing the model to the existing system, analyzing the results, and refining the model [35]. In order to reduce the amount of work needed while simultaneously achieve the best result, it is important to determine the level of details of the real system we need to model in order to achieve a specific goal. This level of details will be determined by the specific problem to be solved, and in such, we need to have a problem based approach to building the model.

When building a model, it is useful to start collecting data early since this is a time consuming task. However, data requirements may surface once the model building process has begun [35], and one should therefore not wait to long before starting the actual building process. The model of the system will result in a great deal of information which will require computation. Because of this, the model must be entered into a computer-recognizable format. One can use either a simulation language or a simulation software/environment, as long as the method is suitable for the task at hand. However, it is stated by Banks et al. ([4]) that the model development time can be greatly reduced if simulation software can be applied.

The process of debugging a model to ensure that every portion operates as expected is called model verification. A common verification technique could be termed *reductio-ad-absurdum*[4], which means reducing a complex model to an aggressively simple case so that we can easily predict what the outcome will be [35]. Once the model is verified we need to validate it to determine that it accurately represents the real system, where a valid model is a reasonably accurate representation based on the model's intended purpose [35]. We will describe the area of verification and validation in more details in Section 2.4.4.

The process of building the model can be managed by asking the following questions [35]:

1. What is the goal of the model?
2. What are the boundaries of the model and what level of detail should be included?
3. Where is the required data?
4. How shall the model be conceptualized?
5. What alternatives will be investigated?

2.4.2 Simulation models

Simulation can be defined as the imitation of the operation of a real-world process or system over time [4].

[4] reducing to the absurd

According to Zeigler et al. ([84]) System Theory becomes central within the area of simulation. System Theory consists of two elements, namely system structure and system behavior. The inner constitution of a system is defined by the system structure. Knowing the system structure allows us to deduce (analyses, simulate) its behavior. System behavior is the systems outer manifestation. Discovering a valid representation of an observed behavior is one of the key concerns of the modeling and simulation enterprise.

In simulation, the system specification formalisms is also important, and is related to the two types of modeling styles used by modelers when building system models, namely continuous or discrete [84].

Based on the two modeling styles, simulation is based on three different types of models, namely continuous-time models, discrete-time models and discrete-event models, which basically differ in their interpretation of time [10][84].

Continuous system simulation

Continuous-time models, hence also continuous system simulation, are based on differential equations [10][84]. Differential equation models do not specify a next state directly, but use a derivative function to specify the rate of change of the state variables [84]. Schemes for solving problems concerning the need to calculate without having computed the input, state, and output trajectories are generally known as numerical integration methods. In continuous system simulation, there are three integration methods, namely Euler, Causal methods and Non-causal methods. However, one problem with this simulation approach, especially with the Euler method, is that the computation time becomes large since the step sizes must be sufficiently small [84].

In continuous system simulation, qualitative analysis of the feedback loops in a system can give insights into its possible behaviors, and the most important behavior to observe is whether a feedback loop is positive or negative [84].

In developing continuous system simulation, one can either use simulation languages based on the Continuous System Simulation Language (CSSL) standard, where the most widespread CSSL language is Advanced Continuous Simulation Language (ACSL), or, one can use block-oriented simulation systems [84]. In block-oriented simulation systems, the modeling is done by coupling together primitive components and elementary functional building blocks [84]. One can then drag and drop blocks into a model to form components in a network.

Discrete Time Simulation

Cellier ([10]) describes that the time axes in this type of simulation is discretized and are commonly represented through sets of difference equations. There is a stepwise mode of execution. That is, at a particular time instant, the model defines how a current state changes to a new state [84]. As the name indicates, time advances in discrete steps (integer multiples of some basic period such as 1 second or 1 year).

Discrete time models have numerous applications and most popular are in digital systems. Discrete time models is frequently also used as approximations of continuous systems [84].

At every time step each component undergoes a "state transition"; which occurs whether or not its state actually changes. It is important to notice that since most often only a small number of components actually change, the Discrete Time Simulation in most situations becomes

inefficient [84].

Discrete Event Simulation

Discrete-event models are usually described by an enumeration of all possible event types together with either a list of times when these events occur or a set of conditions under which they occur [10]. The time axis of this models is, paradoxically, usually "continuous", but this type of simulation differ from the continuous-time simulations by the fact that, in a finite time span, only a finite number of state changes may occur [10].

Events can be caused by the environment (external events) or the component themselves may schedule events to occur (internal events). In the former point the occurrence of events are not under control of the model components itself, while in the latter point the component itself determines their time of occurrence [84].

Since simulation modeling is not done by writing out a dynamic system structure itself, but indirectly, by using system specification formalism [84], Discrete Event System Specifications (DEVS) is of relevance. DEVS was developed for use in Discrete Event Simulation (DES), and provides a hierarchical, modular approach to constructing DES models [84]. DEVS can model systems whose discrete event nature are not immediately apparent and include the means to build models from components. DEVS is most naturally implemented in computational form in an object-oriented framework. DEVS is important not only for discrete event modeling, but also because it affords a computational basis for implementing behaviors that are expressed in the other basic systems formalisms - discrete time and differential equations.

A problem that arises in DES is that of simultaneous events. Several approaches to the problem have been developed [84]. One can let all events undergo their state transition together, which is called Parallel DEVS or one can define a priority among the components. The latter is employed by most simulation packages and in Classic DEVS. A tie-breaking procedure is used, which selects one event to process out of a set of contending simultaneous events. The event scheduling strategy is the most common method. In addition to event scheduling, activity scanning and process interaction, which is a combination of event scheduling and activity scanning, are two other methods of dealing with the issue.

2.4.3 Optimization methods

Simulation optimization is according to Carson et al. ([9]) the process of finding the best input variable values from among all possibilities without explicitly evaluating each possibility. Furthermore, it is stated that the objective of simulation optimization is to minimize the resources spent while maximizing the information obtained in a simulation experiment [9].

Simulation optimization methods can be divided into six categories, namely Gradient Based Search Methods, Stochastic Optimization, Response Surface Methodology (RSM), Heuristic Methods, Asynchronous team (A-Team) and Statistical Methods [9].

Monte Carlo is another class of methods/approaches, which are useful for modeling phenomena's where there is significant uncertainty in the inputs. In optimization, most of the Monte Carlo methods are based on random walks, which is a mathematical formalization of a trajectory that consists of taking successive random steps [81].

Carson et al. ([9]) states that A-team is a process that involves combining various problem

solving strategies such that they can interact in synergy. Furthermore, Carson et al. states that A-team is fast, robust, and inherently suitable for multi-criteria simulation optimization problems, and that A-team represents one of the fastest growing areas of simulation optimization research.

According to Swisher et al. ([72]) the type of simulation optimization techniques appropriate depends on whether the input parameters are continuous or discrete. When the input parameters are continuous, one can choose between both gradient and non-gradient approaches. When the input parameters are discrete and the number of input parameter values is finite and small (under 20) Statistical Methods are appropriate, i.e. Ranking and Selection (R&S) and Multiple Comparison Procedures (MCP), where Multiple Comparisons with the Best (MCB) is the most popular approach. When the input parameters are discrete but the number of input parameter values is infinite or very large, Heuristic Methods are used, e.g. ordinal optimization, Simulated Annealing, Tabu search, and Genetic Algorithms.

Statistical methods can be divided into three methods, namely R&S, MCB and Importance Sampling [25]. One has seen a shift towards discrete sets of input parameter values [72] where R&S and MCPs have gained popularity in simulation optimization [73]. R&S provides the best system design (i.e. it provides the optimal settings for input parameters) while MCPs provide information about the relationships among the designs, and both are particularly well suited for computer simulation [73]. R&S is divided into indifference-zone and subset selection, and where the subset selection is more attractive for discrete-event simulation. Indifference-zone procedures remain the more popular of the two [73]. In contrast to R&S procedures, in which the goal is to make a decision, the goal of MCPs is to quantify the differences between systems' performance [73]. MCPs have according to Damerdji et al. ([17]) the advantage over classical R&S procedures in that the problem is treated as one of inference; the confidence intervals provide information on how close the systems may be to one another. The idea of combining R&S procedures with MCB is appealing to the simulation analyst. Such an approach not only selects the best system with pre-specified confidence, but also allows one to draw inferences about the relationships between systems that may facilitate decision-making based on secondary criteria that are not reflected in the performance measure selected [73].

Evans et al. ([19]) presents a framework for multi-criteria optimization of simulation models. It is suggested that an optimization technique chosen for a particular simulation model should depend upon several important problem characteristics. These characteristics are namely the number of decision variables and criteria, the nature of the response (i.e. output) surfaces (e.g., convex or non-convex), the nature of the response variables (deterministic or stochastic), the run time for the model, and finally, the ability/desire of the decision maker to articulate various types of preference information, concerning tradeoffs between the various criteria.

Optimization methods for rare events simulation

Three examples of optimization methods for rare event simulation have been identified. Heidelberger ([25]) uses Importance Sampling as a method to speed up rare event simulation in Queuing models and Reliability models. Swisher et al. ([72]) describes two other methods, namely ordinal optimization (discrete input parameters method) and Stochastic counterpart algorithm (continuous input parameter method).

It is stated in [25] that in both Queuing and Reliability modeling, (discrete event) simulation

may be the only feasible approach to a solution, although we still need to apply Importance Sampling so that small probabilities can be accurately estimated. Rare events in Queuing models and Reliability models occurs based on very different reasons [25]. In queuing models, the rare event happens because of a combination of a large number of events, none of which are particularly rare, while in reliability models, rare events happen because of the occurrence of only a few events, each of which is itself rare. These are important differences, where each reason requires different importance sampling approaches [25]. In the queuing model situation, so called exponential twisting is required, while in the reliability model situation, failure biasing is required.

Examples of rare events are long waiting times, buffer overflows, system failure of highly dependable computing systems [25]. Another type of rare events of interest are targeted attacks, i.e. specially designed attacks aimed for a specific victim.

2.4.4 Verification and Validation

Verification is the approach to check if a simulator is in error [84]. One attempts to establish that the simulation relation holds between a simulator and a model. There are two general approaches, namely formal proofs of correctness and extensive testing. Recall from Section 2.4.1 that a possible verification approach was to reduce a complex model to an aggressively simple case so that we can easily predict what the outcome will be. Examples of such an approach would be to uncouple parts of the model that interact to see how they run on their own, or to remove all variability from the model, making it deterministic [35].

Validation is how to check if a model is in error [84]. As stated in [35], one of the best validation measures is "Does the model make sense?" Other methods involve obtaining approval of the results by those familiar with the actual process and comparing simulation results with historical data [35].

When talking about validation, one separates between quantitative and qualitative comparison [84]. In quantitative comparison, validation requires comparison of model and source system behavior. In the conventional approach, comparison requires a metric and a tolerance, where the metric provides a numerical basis for measuring "goodness-of-fit" and the tolerance is a positive number that determines when the fit is good enough. When stochastic processes are employed, comparison involves a further consideration since statistical techniques often make assumptions characterizing the stochastic nature of the data sources, and these are models themselves which may not be valid. Although quantitative comparison provides an objective basis for validation, it can miss more qualitative discrepancies or agreements that humans are capable of detecting if provided the right perspectives [84]. Two methods attempting to provide such perspective are visualization and animation. Quantitative comparison is needed to make finer distinctions between behaviors that agree in their basic form, but qualitative comparison can quickly eliminate models that are not even "*in the right ballpark*" [84].

A comprehensive description of model verification and validation is covered by Sargent ([68]). The description illustrates how the different validation and verification (V&V) approaches fits into the "Real-World-to-Simulation-World"-relationship. In this relationship, where System Theory is the link between the two "worlds", Theory validation, Operational (results) validation, Con-

ceptual Model Validation, Specification Verification and Implementation Verification are identified as the V&V approaches.

We will finally in this Section, as we have done in the former Sections, present related work identified within the area of simulation.

In [12], Cohen determines the cost-benefits of security controls by simulating attacks, defenses, and consequences in complex cyber systems such as computer networks. The goal of the modeling process performed in [12] was to generate a set of cause-effect chains that would allow the author to simulate the process of attack and defense. Several classes of threats, attack mechanisms and protective mechanisms were used, and Cohen used a database to associate these with other characteristics such as their impact on integrity, availability, access, and leakages.

The cost estimation, based on ALE, is done by summing both the fixed and a per-use cost of each attack and defense method from each simulation run, although it is argued by Cohen ([12]) that the simulation only needs to assess the per-use costs. Marginal improvements in protection effectiveness might have large enough financial impact to warrant in-depth examination. It is also suggested that response costs may become quite important, which was also identified by Wei et al. ([79]). The ROI in the quality of a defender is non-linear, meaning that with faster detection and reaction, the skills of the defender becomes less critical with regards to success [12].

Landau et al. ([44]) presents a framework for business-oriented modeling of IT infrastructure. The methodology focuses on creating pre-prepared models of standard components by the use of building-blocks. The authors of [44] base their methodology on three fundamental principles, namely that one should

1. only model at the required level of detail
2. model standard components using pre-prepared models
3. automatically deriving the application-specific model details.

The modeling methodology presented in [44] follows a two-stage process, where first a topological IT model is built by assembling building blocks, and secondly, deployment of application-specific automatic model-learning algorithms is applied. The validation and tuning of the model is performed based on IT metrics.

Finally, Popkov et al. ([64]) presents an approach for modeling the IT infrastructure with the focus on an IT cost analysis. This paper is of relevance as it provides a good description of how to model the business processes, and which parameters to consider when determining the affect of IT decisions on the business.

2.5 Challenges

In this final Section we present a set of challenges identified by the authors in our related work. These challenges illustrate some of the larger issues within the combined area of security and simulation, and we will therefore during our thesis attempt to provide insight into these challenges.

1. One of the challenges we face are that we need to have consensus on how to describe a protection system, and that there is no set of commonly accepted metrics upon which to

base a set of measurements to be used for simulation [12]. However, by using KPI and the underling metrics, we might be able to solve this problem.

2. An issue that is stated in [49], is determining the best approach to the valuation of loss of customers and reputation resulting from security problems. The combination of simulation and KPI might be the solution to such an issue.
3. Bartolini et al. ([6]) describes that one needs to link performance optimization metrics with KPI or impact metrics that are meaningful at the business level.
4. Butler ([8]) describes that the security managers he has talked to have asked for an automated tool to support sensitivity analysis which could help to ensure consistency in the evaluation of security decisions. They could use such a tool to quickly enter the new information and see the effects of these changes in their system. This suggests that there is a demand for tools based on simulation.
5. A challenge however, as described by Jacobson et al. ([39]) and Holm et al. ([29]), is that it becomes difficult to get a buy-in of all stakeholders, or otherwise get acceptance of the results from the simulation by others within the management group. Such a challenge could provide an obstacle which will render the process of modeling and simulation useless.

3 Choice of Method

In this Chapter we will determine which scientific methodology we need to select based on our research questions. Furthermore, we will in this Chapter briefly describe our approach for comparing the simulation approach with that of a non-simulation approach. Finally, we will determine which of the simulation methods identified in Chapter 2 is most suitable for our objective of analyzing security decisions effect on the organization.

3.1 Choice of scientific method

Recall from Chapter 1 that our objective is to determine the suitability of combining the simulation approach and the usage of KPIs with regards to determining the effects of a security decision. Hence, we need to identify a scientific method which allows us to determine how suitable this combination is.

Yin ([83]) identifies seven scientific methods, namely case studies, experiments, surveys, histories, the analysis of archival information, interviews, and questionnaires. Depending on the following three conditions, each strategy has their advantages and disadvantages [83]:

1. The type of research question
2. The control an investigator has over actual behavioral events
3. The focus on contemporary as opposed to historical phenomena

In [83], it is stated that if the research question is of a "what" type, and is exploratory, any of the research strategies can be used. On the other hand, if the "what" questions is more in a "how many" or "how much" form, or the research question is of a "who" or "where" type, survey or archival strategies are the best choice. Finally, in answering "how" and "why" questions, which are more explanatory, the preferred research strategies are case studies, histories, and experiments. Table 1, which is taken from [83], summarizes the different research strategies.

Table 1: Comparing different research strategies

Strategy	Form of Research Question	Requires Control of Behavioral Events?	Focuses on Contemporary Events?
Experiment	How, Why?	Yes	Yes
Survey	Who, What, Where, How many, How much?	No	Yes
Archival analysis	Who, What, Where, How many, How much?	No	No
History	How, Why?	No	No
Case study	How, Why?	No	Yes

Case studies are the preferred strategy when "how" or "why" questions are being posed, when the investigator has little control over events, and when the focus is on a contemporary phenomenon within some real-life context [83]. The case study method allows us to retain the holistic and meaningful characteristics of real-life events, such as organizational processes. Based on this information, we choose the method of case study. One of the reasons for this choice is that our research questions falls under the category of exploratory questions. Further, we have little control over the events to be analyzed and the case can also be said to be a contemporary phenomenon within a real-life context.

3.2 Approach for determining the suitability of the simulation approach

In order to determine the suitability of the simulation approach we need to compare this with a non-simulation approach. We will therefore utilize the non-simulation approach described by Helkala et al. in [27]. The comparison will be made based on the same scenario which the simulation approach is applied on, and hence, we are able to compare the processes of conducting two different case studies. The non-simulation approach described in [27] is selected since it conducts a ranking of authentication mechanisms, and since it is constructed within the health care sector. In other words, it focuses on the same areas as we will focus on in our scenario. We will in Chapter 9 describe in more details why we have selected the non-simulation approach found in [27].

As mentioned, both approaches will utilize a scenario which we will construct. It is therefore important to notice that since the involved data collection phase, especially with regards to the simulation approach, is both important and time consuming (see Chapter 4) this would initially suggest that our approach is not suitable given our short time frame available. However, we are in fact able to create the needed scenario since we have access to data which have already been collected over a long time period (see Appendix G).

3.3 Choice of simulation method

Although our third research question defines that we are to apply simulation in our case study, we need to determine which simulation model to choose. Recall from Chapter 2 that three types of simulation models were identified. We will however only consider two of the most used approaches and determine in this Section which of the two we are to apply in our thesis.

We need to select a simulation method that allows us to understand the goals of the organizations and how security controls influence these goals. In such, we need to be able to capture a particular process and determine how the controls affect this process. When deciding which approach is best suited to model a particular problem, the key questions to ask according to Sweetser ([71]) are

1. Which type of model best represents the system under study
2. What questions does the decision-maker wish to address, and
3. For what purpose will the model be used

When deciding which approach to select, one also needs to take into consideration the pro-

vided resources and other constraints which will affect the choice.

3.3.1 Continuous simulation

Continuous simulation, or System Dynamics (SD), have according to Forgia et al. ([22]) been proven to be an effective methodology by which complex systems are modeled based on simple building blocks, i.e. flows, levels and converters. In a different way from DES, that build bottom-up, Forgia et al. ([22]) argues that SD is able to capture complexity from a top down approach that is more suitable for data driven applications. According to Sweetser ([71]), SD is well suited to modeling continuous processes, systems where behavior changes in a non-linear fashion, and systems where extensive feedback occurs within the system. Hence, SD is able to assist organizations in strategy development, analysis of policy options, and analysis of dynamic processes where capturing information flow and feedback are important considerations [71]. SD models often incorporate "fuzzy" qualitative aspects of behavior that, while difficult to quantify, might significantly affect the performance of a system [71].

Some of the criticism to DES is related to the fact that when detailed historical data does not exist, assumptions are highlighted and vetted with the model's users, which can make DES models become "prisoners of the past" [71]. Sweetser ([71]) further argues that since the factors which affect the behavior of the system change over time, the ability of DES models to predict behavior declines. In the absence of historical data, data from the performance of similar systems might be used, but this requires benchmarking data, which might not exist.

3.3.2 Discrete Event Simulation

In the context of Business Process Modeling (BPM), discrete event simulation is the most flexible and powerful tools available [43]. Simulation in the context of process modeling actually refers to discrete computer events simulation according to Laguna et al. ([43]). Furthermore, DES has the capabilities that make it more appropriate to the detailed analysis of a specific, well-defined system [71]. Another benefit with DES is that it is equally well suited to represent qualitative as quantitative models, although it requires that the model as a whole be formulated as a discrete-event model [10]. Finally, as stated earlier, discrete event modeling is an attractive formalism because it is intrinsically tuned to the capabilities and limitations of digital computers [84].

3.3.3 Conclusion

As we have seen, both simulation methods have their benefits, and both have capabilities that we need. The important distinction between the two simulation methods, is that SD more often models abstract, general systems, such as a market for a particular good, while DES models, in contrast, typically have a narrower focus, such as modeling a production line or a call center [71].

Based on the aforementioned arguments, we will adopt the DES method in order to understand how the security decisions influence the goals of the organizations by utilizing KPIs. An alternative could of course be to combine the Continuous and Discrete Event Simulation methods, however, it is much more efficient to adopt a pure event-based approach since one can concentrate on the interesting events only and jump from one interesting event to the next, omitting the uninteresting behavior in between [84].

Since our objective is to compare different system designs, we want to be able to determine the difference in the mean performance and determine which of the system configurations is "best". If we were to compare more than two system design, the Multiple Comparison with the Best (MCB) would have been selected since this approach both have the ability to quantify the difference between systems' and provide information about the relationships amongst the system designs. However, since we are only comparing two system designs, $k = 2$, we will in our case perform a similar, but somewhat simpler approach, by manually comparing of the performance measure based on constructing a confidence interval.

4 Modeling and Simulation Methodology

In this Chapter we will first describe the process of selecting a modeling and simulation methodology to be used in our thesis. We will then describe the methodology used in our case study (Chapter 7) in order to create a model of the scenario (described in Chapter 6).

4.1 Selection of methodology

In order to determine the suitability of the simulation approach for analyzing the effects of a security decision, we also need to utilize a methodology which allow us to conduct the model and simulation process. We have choose to use the methodology described in Banks et al. ([4]) which we determine as a solid source. However, in order to confirm the validity of this methodology, we have compared it with two other methodologies, namely those describe in [46] and [35]. The comparison allowed us to determine if our selected methodology incorporated those elements recognized by others as important.

Table 2: Comparing modeling and simulation methodologies

Simulation steps			
Step	**Banks et al. ([4])**	**Law et al. ([46])**	**Imagine That Inc. ([35])**
1	Problem formulation	Formulation of the problem and the plan of study	Formulate the problem
2	Setting of objectives and overall project plan	Collection of data	Describe the flow of information
3	Model conceptualization	Conceptual model design	Build and test the model
4	Data Collection	Validation	Acquire data
5	Model translation	Construction of the computer representation of the model	Run the model
6	Verification	Verification	Verification
7	Validation	Design of experiments	Validation
8	Experimental design	Production runs	Analyze your results
9	Production runs and analysis	Statistical analysis	Conduct experiments
10	More runs?	Interpretation of the results	Document
11	Documentation and reporting		Implement your decisions
12	Implementation		

The comparison, which can be seen in Table 2, illustrates that all three methodologies includes more or less the same steps. In order to further determine the suitability of the methodology describe in [4], we compared the analyses phase with that described by Centeno et al. ([11]). Based on this comparison, and the fact that all three methodologies and process incorporate the similar steps, we have confidence in our choice of basis for the methodology used in this thesis.

4.2 The modeling and simulation methodology applied in the thesis

The methodology described in this Section is a simplification of the methodology described by Banks et al. ([4]). As mentioned in the former Section, this methodology is used since it was

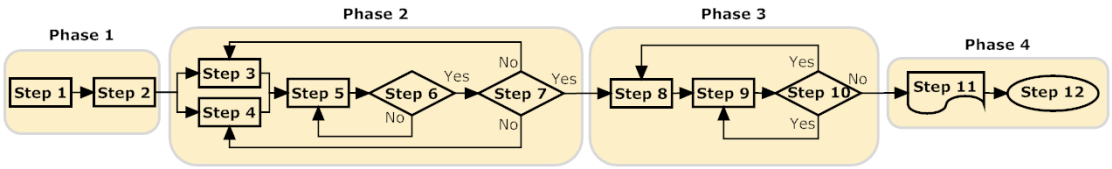

Figure 1: Categorization of steps found in modeling and simulation methodology

recognized as a solid foundation for the modeling and simulation process. The twelve steps found in this methodology, which was listed in Table 2, have been categorized into four phases, as illustrated in Figure 1.

We will in this thesis only utilize the first three phases, and have chosen to divide the steps included in these phases into the following five steps.

1. Preparation (Phase 1)
2. Design and data collection (Phase 2)
3. Model building (Phase 2)
4. Simulation parameter estimation (Phase 3)
5. Analyzing the simulation output (Phase 3)

If the reader is unfamiliar with simulation, the reader is strongly advised to read Appendix E before reading this Chapter in order to get a deeper insight into simulation. All variables used in equations presented in this Section are explained in Appendix C.

4.2.1 Preparation

The preparation phase lays the foundation for the whole simulation process by focusing on determining the scope and the objectives of the project.

The objectives indicate the questions to be answered and a determination should at this point in time be made concerning whether simulation is the appropriate solution for the problem formulation and objectives given. The issue of whether simulation in fact is an suitable approach is an important issue, and one which we will focus on in this thesis.

4.2.2 Design and data collection

Model conceptualization is, as stated in [4], as much art as science. However, there are some guidelines that we can follow. The essence of model conceptualization is having the ability to abstract the essential features of a problem, to select and modify basic assumptions that characterize the system, and then to enrich and elaborate the model until a useful approximation results.

It is recommended in the methodology to start with a simple model and build toward greater complexity. However, it is emphasized that the complexity should not exceed what is required in order to accomplish the purposes of the models intentions. Another important element in the model conceptualization is involving the model user. This enhances the quality of the resulting model and increases the confidence of the model user in the application of the model.

There is a constant interplay between the construction of the model and the collection of the needed input data, and the required data changes as the complexity of the model change [4]. The methodology emphasizes the importance of beginning the data collection early, as it is a time consuming task. Although the objectives of the study dictates the kind of data that needs to be collected, historic data, where available, should always be collected as this can be used to validate the simulation model.

Verification, which is a difficult task due to the complexity, is performed by correctly representing the input parameters and logical structure of the model in the computer. Common sense is for the most part applied in completing this step. Although verification, and the later described validation, are crucial elements in the entire process, our objectives allows us to skip this time consuming step. That said, we need to take such steps into account when determining the suitability of the simulation approach later on.

4.2.3 Model building

Based on the collected data and design from the previous step, the model can be entered into a computer-recognizable format and be constructed. In specific, the model which will be constructed is based on a state/action model. This represents a system that responds to an event by transitioning to another state. One of the reasons for creating such type of model is that it helps us gaining the correct point of view when running the simulation. That is, we are able to determine how each object interacts with the activities that are identified.

The validation step is an iterative process which usually is achieved through the calibration of the model. The process is repeated until model accuracy is judged acceptable, e.g. by checking whether the simulation model replicate a specific system measure. This is perhaps the most crucial point in the entire process, as an invalid model is going to lead to erroneous results, which, if implemented, could be costly and result in a wrong decision [4]. Due to our earlier mentioned objectives, we will not conduct a thorough validation process. However, when determining the suitability of the simulation approach, we need to take into considerations that such thorough processes are needed.

4.2.4 Simulation parameter estimation

The simulation parameter estimation process consist of determining how to handle the randomness of the model in such a way that the comparison later on will yield the best results, and also making sure that we have confidence in the results that are presented. In other words, proper statistical analysis is required as the output variables are estimates that contain random errors.

For each system design, i.e. passwords and smart cards, that is simulated, decisions need to be made concerning the length of the warm up period, the length of simulation runs, and the number of replications to be made of each run. Based on the analysis done during this estimation process, one has to determine whether additional runs are needed.

Random number streams

As described in Appendix E, random number streams in a model can either be incorporated by *independent sampling* or *Common Random Numbers (CRN)*. In Appendix E it was stated that when comparing different system designs, e.g. implementing smart cards instead of using pass-

word as the choice of authentication mechanism, the use of CRN has some obvious benefits with regards to added precision in the comparison. Because of this, we have also chosen to implement CRN in our model. However, as described in Appendix E, the implementation of CRN is model dependent, i.e. for any type of scenario or problem there are many ways of implementing CRN. However, certain guidelines are given which will make CRN more likely to yield a positive correlation [4]:

1. Dedicate a random-number stream to a specific purpose, and use as many different streams as needed. In addition, if several replications are used, assign independently chosen seeds to each stream at the beginning of each replication.
2. In systems with external arrivals, as each entity enters the system, the next inter arrival time is generated. Then immediately all random variables needed by the arriving entity and identical in both models are generated in a fixed order and stored as attributes of the entity, to be used later as needed. Dedicate one random-number stream to these external arrivals and all their attributes.
3. For systems having an entity performing given activities in a cyclic or repeating fashion, assign a random-number stream to this entity.
4. If synchronization is not possible, or if it is inappropriate for some part of the two models, use independent streams of random numbers for this subset of random variables.

Determining simulation setup

The type of system to be simulated, either terminating or non-terminating, also determines the method used for determining the simulation setup (Appendix E). Since we in Chapter 7 identify that the scenario (Chapter 6) consist of a non-terminating system, we will only describe how we are to determine the simulation setup for such a system. However, for completeness, the steps involved in determining the simulation setup for terminating systems are found in Appendix F.

Determining simulation setup for non-terminating systems

The process of determining the simulation setup for non-terminating systems is somewhat more comprehensive than with terminating systems. In order to answer the fundamental question "for how long should the simulation be run?", two critical issues needs to be addressed, namely achieving steady state conditions and obtaining statistically independent observations [11]. When these issues have been addressed, we can obtain the confidence interval in similar manners as described for terminating systems. The selection of sample size, or the simulation end time, T_E, is a design choice, and not inherently determined by the nature of the problem, as is the case with terminating systems.

We determine the simulation setup with the following steps [4]:

1. Establish the measure of performance, θ, for the analysis, e.g. the mean time in the system. The selection of the measure of performance is based on the KPIs of interest.
2. Decide the type of confidence that we seek, i.e. α

3. Run the model for a short simulation length[1]. The individual observations must be saved as they are to be used later on.
4. Visually determine the warm up period with the use of cumulative average on the sample data (the observations). If the output does not stabilize, run the simulation for a longer period until such stabilization is observed.
5. It is recommended to collect at least 10 times as much data as is deleted, i.e. 10 times as much data as is observed during the warm up period. Should the number of observations collected during step 3 not yields such amounts of data the simulation length needs to be increased.
6. Based on the collected observations, create batches by applying the *batch means method*[2]:
   (a) Obtain the output data and delete output collected during the warm up period.
   (b) Create up to $k = 400$, but at least 100, batches with the remaining data, and compute the batch means, $\overline{Y}_j$.
   (c) Estimate the sample lag-1 autocorrelation[3] of the batch means as

$$\widehat{\rho_1} = \frac{\sum_{j=1}^{k-1}(\overline{Y}_j - \overline{Y})(\overline{Y}_{j+1} - \overline{Y})}{\sum_{j=1}^{k}(\overline{Y}_j - \overline{Y})^2} \tag{4.1}$$

   (d) Check the correlation to see whether it is sufficiently small
      - If $\widehat{\rho_1} \leq 0.2$, then rebatch the data into $30 \leq k \leq 40$ batches, and form a confidence interval using k - 1 degrees of freedom for the t distribution and estimate the variance of $\overline{Y}$ (4.2).
      - If $\widehat{\rho_1} > 0.2$, then extend the replication by 50% to 100% and go to Step 2. If it is not possible to extend the replication, then rebatch the data into approximately $k = 10$ batches, and form the confidence interval, using k - 1 degrees of freedom for the t distribution and estimate the variance of $\overline{Y}$ (4.2)

$$\overline{Y} - t_{n-1,1-\alpha/2}\frac{\sigma}{\sqrt{n}} \leq \theta \leq \overline{Y} + t_{n-1,1-\alpha/2}\frac{\sigma}{\sqrt{n}} \tag{4.2}$$

   (e) Examine the batch means for independence by performing the correlation hypothesis test for zero autocorrelation
      - First, compute the test statistic

$$C = \sqrt{\frac{k^2-1}{k-2}}\left(\widehat{\rho_1} + \frac{(\overline{Y}_1 - \overline{Y})^2 + (\overline{Y}_k - \overline{Y})^2}{2\sum_{j=1}^{k}(\overline{Y}_j - \overline{Y})^2}\right) \tag{4.3}$$

[1] What a short simulation length is depends on the particular system to be modeled, but about 10000 time units can be a initial simulation length

[2] Although no widely accepted and relatively simple method for choosing an acceptable batch size m (or equivalently choosing a number of batches k) exist, we apply the general strategy of [4] based on general guidelines. Although the procedure describe above is conservatism by design - the cost of an incorrect decision is typically much greater than the cost of some additional computer run time.

[3] Autocorrelation is described in more details in Appendix E

- If $C < z_\beta$ then accept the independence of the batch means, where β is the Type I error level of the test (such as 0.1, 0.05, 0.01).
- Otherwise, extend the replication by 50% to 100% and go to Step 2. If it is not possible to extend the replication, then rebatch the data into approximately $k = 10$ batches, and form the confidence interval, using k - 1 degrees of freedom for the t distribution and estimate the variance of $\overline{Y}$

After completing these steps for both systems that are to be compared, the simulation setup fulfills our demands for reliability, and we can hence compare the two systems. That is, we can compare the implementation of smart cards against the implementation of passwords.

4.2.5 Analyzing the simulation output

Because the output from the simulation run contain random variation, statistical analysis is needed to discover whether any observed differences are due to differences in design or merely to the random fluctuation inherent in the model. Having implemented Common Random Numbers (CRN), we have however reduced the variance of the estimated difference of the performance measure since we are utilizing the same random numbers when simulating the two alternative system designs.

We assume that the appropriate steps described earlier in this Chapter have been performed before the comparison is done. Note that, when comparing the two systems[4], the mean performance measure for system i will be denoted by θ_i *(i = 1,2)*. The process for determining the difference between the two systems is as follows [4]:

1. Collect the same number of observations from the two simulated systems
2. Compute the differences

$$d_j = \theta_{1j} - \theta_{2j} \tag{4.4}$$

where d_j is the individual differences and a *m. v. u. e.*[5] of δ, i.e. the true difference between the two systems. θ_{1j} and θ_{2j} are the *j*th observation from system 1 and 2 respectively.

3. Compute the average difference, $\overline{d}$, (4.5), the sample variance of the difference (4.6) and the standard error of $\overline{d}$ (4.7)

$$\overline{d} = \frac{\sum d_j}{n} \tag{4.5}$$

$$\sigma_d^2 = \frac{1}{n-1}\sum_{j=1}^{n}(d_j - \overline{d})^2 \tag{4.6}$$

$$s.e._{\overline{d}} = \frac{\sigma_d}{\sqrt{n}} \tag{4.7}$$

[4]That is, smart cards and passwords
[5]Minimum Variance Unbiased Estimator

4. Compute the confidence interval around $\overline{d}$, i.e. utilize Equation 4.2
5. Determine, based on the computed confidence interval, whether there are significant difference between the two systems. If we assume that having a smaller mean is considered best, then
   (a) If the confidence interval is totally to the left of zero, then there is strong evidence for the hypothesis that $\theta_1 - \theta_2 < 0$, which implies that system 1 is better than system 2
   (b) If the confidence interval is totally to the right of zero, then there is strong evidence for the hypothesis that $\theta_1 - \theta_2 > 0$, which implies that system 2 is better than system 1
   (c) If the confidence interval contains zero, then, in the data at hand, there is no strong statistical evidence that one system design is better than the other.
6. Determine if the confidence interval calculated is practically significant.

The last step found in the process is added for a couple of reasons. While statistically significant answers the question "is the observed difference larger than the variability?", practically significant answers the question "is the true difference large enough to matter for the decision we need to make?". Therefore, should this last step have not been included and given the situation of the confidence interval containing zero, more data would have been needed in order for a conclusion to have been made. However, by including this last step, we might still be able to draw some conclusions about the output by determine what the practical significant differences in the mean performance is. In addition, if we had not including the step of determining the practical significance when making a decision, we may reach the conclusion that $\theta_1 > \theta_2$, and decide that system 1 is better[6], although the practical difference is to small to actually notice, e.g. for a customer. In such situations, although we have reach a conclusion with regards to the statistical difference, the added benefits might not be worth the cost of replace system 1 with system 2, or if none of the systems are implemented, one might need to make the final decision based on other elements. Finally, it is important to remember that should the result of the analysis yield that there are no statistical significant difference, it can in fact be because there are no difference between the systems with regards to this specific measure of performance.

4.3 Discussion

The modeling and simulation methodology applied in this thesis is based on the 12 step methodology described by Banks et al. ([4]). Since it is important that the applied methodology follows commonly recognized processes and steps in the field of modeling and simulation, the methodology has been compared to other modeling and simulation methodologies. When compared to the 10 step methodology described in [46] and the 11 step process described in [35], it was concluded that all three approaches included the same central elements as those described in this Chapter. In order to gain additional confidence in the output analysis phase, this particular phase was compared to the simulation output analyses described in [11]. Since both analysis include the same approaches, and based on the similarities of the aforementioned methodologies, we

[6]Provided that a larger value is considered best

have confidence in that the methodology described in this Chapter can be applied for the purpose of creating a model and conducting the simulation. Therefore, by following the modeling and simulation methodology described in this Chapter, we should be able to determine if simulation is a suitable approach for determine the effects of security decisions on the organization.

5 Simulation Environment

In this Chapter we will describe the selection process conducted in order to determine which simulation environment we should use, and we will briefly introduce the simulation environment which is selected based on this process.

5.1 Selection of simulation environment

Although many different simulation environments exist, they are all intended to assist us in the modeling and simulation process. In order to select a simulation environment which we can utilize in this thesis, we will compare different simulation environments against a set of requirements identified in [4] which we will describe below.

In total, eleven different simulation environments were initially too been considered[1]. However, only five of these, namely AnyLogic, ExtendSim, FlexSim, Micro Saint and OMNeT++, were actually tested as these where the only environments of the total eleven that was possible to test[2].

The five tested simulation environments was considered based on the selection criteria described by [4]. More specific, the different environments have been considered based on their price, ease-of-use, applicability for the task at hand[3], ability to combine graphical model-building with customization with the use of programming and, finally, support (either training, good documentation or other forms of methods in order to solve problems). In addition, as the methodology is aimed at being able to utilize independent of the type of organization, it is important that the simulation environment allows us to model more general models, which makes it possible to draw conclusions with regards of the usability of the methodology across organization types. That is, we want a simulation environment which is not area specific.

In Table 3 we summarize each simulation environments results based on the identified criteria's. It is important to notice that the properties of a simulation environment is compared in relation to each other, e.g. if the price of a simulation environment is considered "high", this is compared to the other four environments.

Out of the five tested simulation environments, AnyLogic, ExtendSim and OMNet++ best suited the criteria's. However, after an internal process, AnyLogic, due to economical reasons, was left out. ExtendSim was considered the best choice of the two remaining environments, but where only possible to choose due to their *Research Grant Program* which has allowed us to use the environment free of charge during the thesis.

It should be noted that the methodology presented in Chapter 4 is not simulation environment

[1] AnyLogic [74], Arena [36], AutoMod [16], ExtendSim [34], FlexSim [33], Micro Saint [70], OMNeT++ [78], ProModel [14], QUEST [13], Simul8 [15] and WITNESS [45]

[2] They either had trial versions available or was free of charge

[3] The simulation environment must support DES and should otherwise not include functions or properties which make the process of modeling not applicable

Table 3: Comparing simulation environments

Simulation environment	Price	Ease-of-use	Applicability	Combining drag-and-drop with customization	Support	Area specific
AnyLogic	High	Yes	Yes	Yes	Yes	No
ExtendSim	Free[1]	Yes	Yes	Yes	Yes	No
FlexSim	High	Yes	No[2]	Yes	Yes	Yes[3]
Micro Saint	High	Yes	Yes	Yes	Yes	No
OMNet++	Free	Yes	Yes	Yes	Yes	No

[1] Free of charge during the thesis based on *Research Grant Program*, [2] Unnecessary advanced for our purpose (3D modeling), [3] The "general purpose simulation package" was not considered sufficiently general for our purpose

specific, and hence, several, if not all, of the simulation environments mentioned in this Section, as well as other simulation environments, should in practice be able to utilize the methodology.

5.2 Selected simulation environment: ExtendSim

We will in this Section give a short introduction to ExtendSim, the selected simulation environment.

ExtendSim supports all three main types of simulation, namely continuous, discrete event and discrete rate. In order to create the model and perform the simulation, we utilize ExtendSim's block library and its internal programming language, called *ModL*.

The block library contains several different types of "blocks", all serving the purpose of creating and processing *Items* and *Values* during the simulation. *Values* are usually used to change the properties of a *Item* or to change the behavior of the model during simulation. Both *Items* and *Values* are created and processed by different types of blocks, where we in our model only utilize a small portion of the blocks found in ExtendSim's block library. A short description of the blocks utilized in our model is found in Appendix D.

ExtendSim's internal programming language, *ModL*, is essentially *C++* with some enhancements and extensions to make it more robust for simulation modeling. *ModL* is used amongst other things to determine the functionality of the "blocks" and one can hence also create own blocks. In some of the blocks found in the block library, one can determine how the "block" handles *Items* and *Values* with the help of *ModL*. One can also with the utilization of *ModL*, communicate directly with the integrated database. In our model, the main usage of *ModL* is for the creation of logical functions[4] and for communicating directly with the database, as the example below illustrates. In this example, the input value, either 10 or 11, determines what value is written to a specific database record determining what state a doctor is in, either knowing his password or having forgotten it. Since we only use *ModL* for more general purposes and since the syntax is very similar to that of *C++*, we will not go into more details into *ModL*. For more information about *ModL*, see [35].

```
if(inCon0 == 10)
DBDataSetAsNumber(9,6,2,1,1); // Know_Pass State = 1
else if(inCon0 == 11)
```

[4] AND, OR, if/else etc.

```
DBDataSetAsNumber(9,6,2,1,0); // Know_Pass State = 0
```

As we are working with a stochastic model, we also need a pseudo random number generator in order to generate uniform random numbers used in the distribution functions. ExtendSim uses the "*minimum standard*" random number generator developed by Lewis et al. ([47]) with the new coefficients described by Dwyer et al. ([18])[5] to generate the uniform random numbers used in the distribution functions. These 32-bit functions are seed-based and update their seed after being called [35].

5.3 Discussion

By utilizing a set of criteria's described in [4], the confidence in the selection process increases compared to determining such criteria's our self. However, there are some issues with regards to the selection process conducted in this Chapter. First of all, additional sources should have been used in order to increase the confidence in the selection criteria's them self. Furthermore, since the evaluation of how each simulation environment fulfills the identified criteria's is a subjective process, there are room for interpretation. Because of this, a different simulation environment could have been selected by another person. Finally, it would have been desirable to compare all eleven identified simulation environments since this might have affected the outcome.

However, it is important to notice that the selection of simulation environment should not affect the decision with regards to the simulation approach's suitability in determining the affects of security measures. Any environment specific issues emerging during the case study (Chapter 7) affecting this question must be treated accordingly.

That said, the selection of simulation environment could have great influence with regards to reducing the work load for the modeler, and also increase the success of the application of the methodology described in Chapter 4. For instance, a simulation environment could simplify the process of analyzing the simulation output, and it could also reduce the possibilities of human error during this phase.

[5] Alternatively, for backwards compatibility issues, ExtendSim also enables the use of the pseudo random number generator described in [69]

6 Scenario

In this Chapter we will describe a health care specific scenario which we will later use in a case study in order to answer our research questions. In the scenario, we identify a patient treatment process found within a Norwegian somatic hospital[1] (in Norwegian: Somatisk sykehus) and the hospitals KPIs. Furthermore, in the scenario we identify the processes and KPIs related to the introduction of smart cards and passwords as an authentication mechanism within a hospital. In specific, the scenario further focuses on doctors and adversaries, and the relevant business activities and security threats relevant for these objects. The scenario, which we will describe in details later in this Chapter, is illustrated in Figure 2.

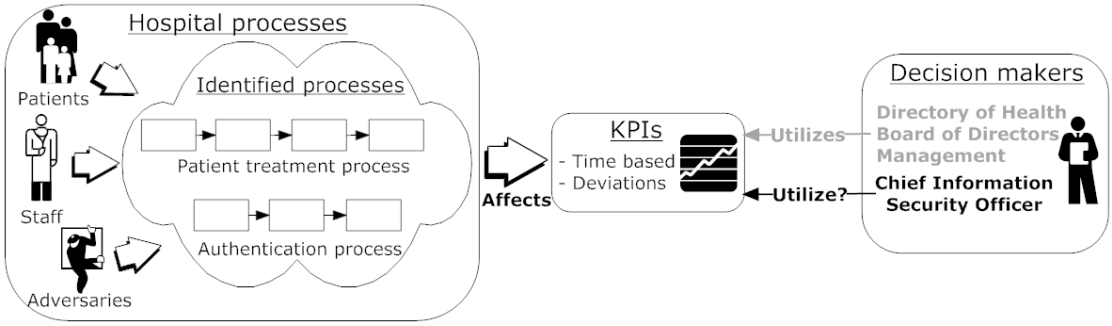

Figure 2: Illustration of health care specific scenario

After presenting the background for the scenario, we will describe the actual scenario in details. The presentation of the scenario is, for clarifications purposes, divide into an environment specific Section and a security related Section. In this context, *environment specific* relates to the actual business process which we have based this scenario on. The structure within each of these sections is similar, where we first determine the scope by identifying relevant process and objects, before describing the identified KPIs. In the security related section, we then identify relevant security vulnerabilities, threats and their consequence. Finally, we will in each of the sections describe the input data which have been collected. It is important to notice that, although we present the sections separately, similar elements have been collected simultaneously, e.g. all KPIs, both environment specific and security related, have been collected at the same time.

Due to the technical nature of the terminology used in the health sector, we have chosen to include the Norwegian translation of words that are not evident. The complete list of translations can be found in Appendix B.

[1]Somatic hospitals, also called acute, general or medical and surgical hospitals, can be defined as ordinary hospitals which provide special treatment for physical diseases and injuries. For the remainder of this thesis, we will refer to somatic hospitals when using the word hospital.

6.1 Scenario background

The health care specific scenario have been developed in collaboration with Akershus university hospital (Ahus), Buypass and the Rheumatism Hospital[2]. In addition, other sources have also been utilized in order to create the scenario, e.g. laws and regulations, and the official web sites of the Ministry of Health and Care Services.

The patient treatment process described in this Chapter has been identified within the Rheumatism Hospital. The key reason for choosing the Rheumatism Hospital as our basis for the patient treatment process is that these processes are small enough to be manageable but still large enough to include all elements needed in order to determine the suitability of the simulation approach. Considering our short amount of time available, the fact that information about the processes was available also became an important decision making argument.

The smart card authentication mechanism is developed by Buypass, who, in collaboration with Ahus implemented the mechanism at Ahus. The daily maintenance and operations of the authentication mechanism are conducted by Ahus. The ID office located at Ahus functions as the contact point for the employees if problems with the authentication mechanisms occur.

There are several reasons why we have chosen to test the suitability of the simulation approach on a process found within the health sector. One reason is the fact that the health sector has a high maturity level with regards to information security. The second reason is that the needed input data for creating the model in fact have been available for us to utilize.

Since our intentions with the scenario is to determine the suitability of the simulation approach in general, and not performing a analysis of an actual security decisions for a specific hospital, some simplifications and assumption have been applied in the scenario. These simplifications and assumptions should however not affect our objective, namely to determine if the simulation approach is suitable for determining the effects of a security decision. Because of the simplifications and our objective, it is important to notice that the data used for the scenario, although mostly based on figures received from real world systems, will not be attempted to be validated or verified. Therefore, the results from our analyses of the scenario cannot be used directly for determining which of the two security measures are considered best for the hospital in question.

6.2 A health care specific scenario

Recall from Chapter 2 that it was stated that security controls must be considered in the context of the business in which they operate, and that security controls are designed to ensure that an organization meets its confidentiality, integrity, and availability objectives. In such, by measuring the KPIs, we will determine how the security controls influence the delay of a business activity, the availability of medical records, and the security aspects of the organization. Ensuring the confidentiality, integrity and availability objectives is perhaps even more important in a health care environment since the consequences of not fulfilling the demands could be catastrophic, e.g. full unauthorized access to medical records [57].

Recall that the scenario is, for clarifications purposes, divide into an environment specific

[2]The Rheumatism Hospital is a small, private, non-commercial, Norwegian hospital situated at Lillehammer which specialized in rheumatic diseases

Section and a security related Section but that the structure within each of these sections is similar.

6.2.1 Environment specific elements

We will in this Section first determine which steps the patient treatment process consists of, while we in the next Section will determine how the different security measures interact with this business process.

The patient treatment process

Based on flow charts made available by the Rheumatism Hospital, found in Appendix H, we have been able to identify the essential steps included in a patient treatment process. The steps, illustrated in Figure 3, are defined in this thesis as *Business Activities* (BAs).

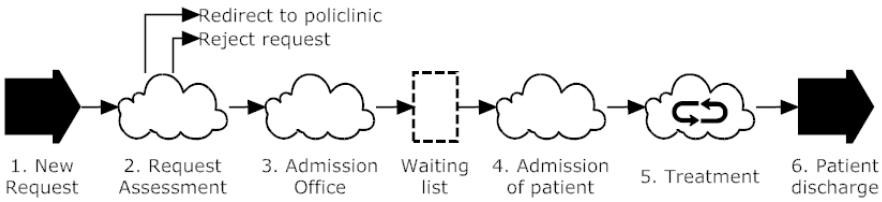

Figure 3: Illustration of the patient treatment process

In the patient treatment process, when new requests for treatment arrive (step 1) they are assessed by doctors (step 2). These requests are rejected, redirected to the polyclinic, or approved and forwarded for admission. Those requests that are forwarded are processed (step 3), and the patient is placed in a waiting list. When the patient is due for treatment, he or she is admitted to the hospital (step 4). Once admitted, the treatment process (step 5) is repeated until the patient is either discharge or dies (step 6).

Several different types of employees are involved in this patient treatment process, including doctors, nurses, secretaries and administrative personnel. In the scenario, doctors and nurses, which are authorized to access medical records through the application DIPS (In Norwegian: Distribuert Informasjons- og Pasientdatasystem i Sykehus), are found in the category "Health related personnel". Within the administrative personnel, we find ID office workers, which for example are authorized to issue new smart cards. In addition, in the process, *outsiders* and *insiders* are also relevant, where insiders relates to doctors who access medical records without service needs (snooping). We could have further specified the difference between similar employees if this was desirable or necessary to answer specific questions, e.g. whether there is a difference between surgical doctor and general practitioner with regards to the final results. We will however not determine such differences in our scenario.

In order to simplify the scenario, we will only focus on doctors, and only on two of the BAs that is performed by a doctor. In specific, we will focus on the BA of *approving documents*, which occur during or after step 2, 4, 5 and 6, and the BA of *patient treatment* (step 5). As one would expect in real life, we will prioritize the *patient treatment* requests higher than *approving documents* requests. We will however not include functions such that a doctor who currently is

processing an *approving documents* request aborts this should a *patient treatment* request arrive.

Since the number of doctors at Ahus have been stable the last couple of years ([30] and [31]), we have not included a distribution of the number of doctors which leaves the hospital and the number of doctors that are hired. Because of this, we simplify the scenario even further with regards to the turnover of employees. Based on this, in our model, we have a constant number of doctors. In specific, 12 doctors are included in the model. The number of doctors is not chosen for any specific reasons.

Another simplification which we will make is that the 12 doctors included in the model will each have a 12 hour work shift. The purpose for doing such is that this will reduce the complexity of the model considerably, and we only need to handle two work shifts. Although this naturally would not have been possible in real life, this still allow us to simulate a hospital that always has employees at work and where business activities are conducted at all hours of the day. We have deliberately chosen to not include elements such as different work load throughout a day[3], again to simplify the model. We do however believe that these, rather large, simplifications does not affect the outcome as both types of security measures will be affected equally to the simplifications, and the simplifications does not affect our objective.

Figure 4 summarizes the model specific figures used. As earlier mentioned, these figures are not selected for any particular reasons and are scenario specific.

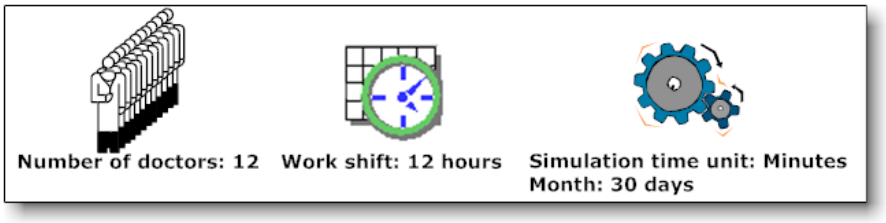

Figure 4: Summarize of model specific figures used

Environment related KPIs

Several sources have been utilized in order to determine which KPIs that is relevant with regards to our scenario. We have had conversations with several people within the health sector, including Rolf Kulstad, Halvor Sandodden, Ellef Mørk and Beate M. Huseby. Furthermore, the web sites of both the Ministry of Health and Care Services, and the Norwegian Directory of Health have been used. Finally, several different documents, including [55], [42], [51] and [28], have been utilized.

Based on the sources, four different time related KPIs have been determined as relevant for our scenario, as listed in Table 4. We describe these KPIs in more detail below.

[3]Normally, the busy hours are between 10 AM and 7 PM during the weekdays [29]

Table 4: Environment Specific KPIs identified

Identified KPIs	Description	Source
Number of assessment warranty breaches	The KPI increases should a patient's referral not be assessed within 30 days of its arrival.	[77]
Average waiting time	The KPI describe the average waiting time for patients. The requirements with regards to the KPI is that the waiting time for patients with legal right to necessary health care must not increase.	[2], [50]
Number of time limit breaches	Should a patient not receive treatment within the time limit which is set by a doctor, a time limit breach occurs. The share of such time limit breaches must be reduced.	[52]
Length of stay	The length of stay can be thought of as the through-put of patients in a hospital and is defined as the number of days a patient with accommodation uses a bed. The length of stay should be as short as possible.	[2]

Number of assessment warranty breaches (in Norwegian: Antall brudd på vurderingsgarantien)

According to Section 2-2, 1st paragraph in the Patients' Rights Act [77] (*in Norwegian: Pasientrettighetsloven*), "*[a patient] is entitled to have his or her health condition evaluated within 30 working days of receipt of the referral*". The KPI is hence increased should a patient's referral not be assessed within 30 days of its arrival.

Average waiting time (in Norwegian: Gjennomsnittlig ventetid)

According to the Patients' Rights Act, section 2-1, 2nd paragraph [77], "*the patient is entitled to receive necessary health care from the specialist health service. This right only applies if the patient can be expected to benefit from the health care, and the costs are reasonable in relation to the effect of the measure*".

Patients with legal right to necessary health care consists of about 60 percent of the total number of patients and has an average waiting time of 68 days. The remaining patients (about 40 percent) have an average waiting time of 84 days [2]. The requirements with regards to the KPI are that the waiting time for patients with legal right to necessary health care must not increase.

The KPI describe the average waiting time for patients[4] who have scheduled treatments, which is about 35 percent of the patients [50]. The remainder 65 percent of the patients treated is found in the urgent care category and is not covered by this KPI.

Number of time limit breaches (in Norwegian: Antall fristebrudd)

If the patient is entitled to receive necessary health care from the specialist health service, then, according to the Patients' Rights Act, section 2-1, 2nd paragraph [77], "*the specialist health service shall set a time limit within which, when justified for medical reasons, a person with such a right shall receive necessary health care*".

Should a patient not receive treatment within the time limit which is set, a time limit breach occurs. The share of such time limit breaches must be reduced, according to the 2010 mission

[4]It is not actually patients that are listed in a waiting list, but rather referrals or applications for approval. Because of this, a person can in fact be listed several times in the waiting list. We will however use the term patients as this is most common.

document for southeast health division [52].

Length of stay (in Norwegian: Liggetid)

The length of stay can be thought of as the through-put of patients in a hospital and is defined as the number of days a patient with accommodation uses a bed. It is important to differentiate between patients receiving treatment for one or several days, but without accommodation, and those patients which are considered by this KPI (which have accommodation)[5]. The length of stay is computed by subtracting the entry date from the discharge date, while the average length of stay is computed by dividing the number of discharged patients with the total sum of length of stay within a particular group of discharged patients. The length of stay should be as short as possible [2].

Since we are only to test the suitability of combining the simulation approach and the usage of KPIs, and not perform a specific analysis of a security measure, we have chosen to make some simplifications with regards of the KPIs to be analyzed. Since all the identified KPIs are time related, we chose to categorize all four KPIs into one KPI, called *time related KPI*. When analyzing the output of the model, we will only focus on how the security measures affect the time used on a business activity. Based on the goals of the individual KPIs, the security measure which yields the lowest *time related KPI* value is considered the "best" security measure.

Although this simplification reduces the level of detail in the analysis, the model still provides us with information with regards to how the security measure affects the time used on a business activity. Should we therefore observe an increase (or decrease) in the time used on a business activity when a specific security measure is applied, we will assume that such effects also would affect the individual time related KPIs.

Input data collected

We will in this Section present the environment specific input data which will be used in the model. The original data received from Ahus and the Rheumatism Hospital, in which the input data have been based on, can be found in Appendix G. In Appendix G, we also describe how the data have been processed into a uniformed scale. The data have been gathered based on the identified patient treatment process and the identified KPIs described earlier. It is important to note that all figures presented in this Section are adjusted based on the scenario, i.e. the figures are based on 12 employees. The "per-user" figures, which the input data is based on, are found in Appendix G.

The data listed in all Tables both in this Section and the next, have either been collected based on Subject Matter Expert (SME) opinions, Documented (D) data, Estimated (E) data, or a combination of these sources. Preferably, documented data[6] is used. When such data is not available, SME data will be used. The difference between SME and estimated data is that the former are estimates made by persons which specific insight into the area of interest, while estimated data is estimations and assumptions made by the author. We have, to our best abilities, avoided to use estimated data since much uncertainty exist in such data. However, where no

[5]That said, one can actually have length of stay for a patient where the discharge date is equal to the entry date since one includes all planned accommodations, and the treatment sometimes takes shorter time than originally estimated.

[6]Documented data are for example registered requests of new PIN code at Ahus.

available data exists, estimated data have been used[7]. For a better quality in the data, rather than purely estimated data, a combination of several types of sources has also been used. With such figures we need to make some assumptions with regards to how the figures can be used as input data in our model. Such data will hence be based on documented data, or SME data, but have been altered based on estimation done by the author. It is assumed that such data will be better than purely estimated data.

Table 5: Input data - Environment Specific Distributions

Parameter	**Time Between Arrivals (TBA) / Duration**			**Source**
	Minimum	**Maximum**	**Most likely**	
TBA of patient treatment request	12 min	15 min	13 min	E
Duration of treatment request	50 min	70 min	60 min	SME
TBA of approve document request	40 min	120 min	60 min	E
Duration of approve request	2 min	5 min	3 min	(SME)/E

As we can see from Table 5, we have determined that a new patient treatment request occurs at a rate of 13 minutes and takes 60 minutes to process in general, while a approve document request occurs at a rate of 60 minutes and takes 3 minutes to process. The time between arrival (TBA) of patient treatment requests and approve document request are estimated by the author. The estimation is based on the duration of each type of request, which is based on data from the Rheumatism Hospital, and one single work shift of a doctor. As we wish to determine how the security measures affect the business activities, and are less concerned about the accuracy of the number of requests that arrive, our goal have been to create sufficiently many requests such that each doctor always will have work to do. We have chosen to do this in order to simplify the analysis of the output, although this is not a realistic situation, e.g. a doctor needs breaks during a work day. It becomes easier to detect the difference between the two security measures with respect to the increased (or decreased) time used on a business activity when not incorporating that doctors in the model are available for a certain amount of time. Although it is possible to incorporate such features in the model, we chose not to do so since the simplification will affect each security measures equally, and we save time as we can create a simpler model.

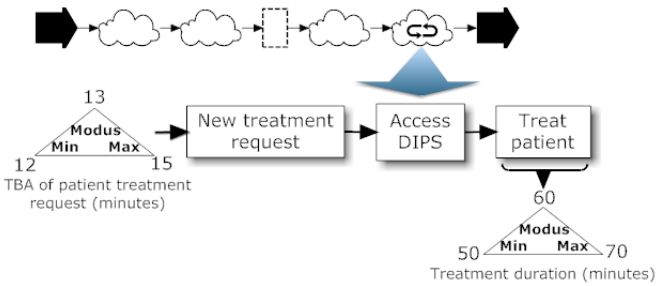

Figure 5: Illustration of the treatment process with input data

[7]Given our goals for creating the model, we are not depended on having data that are based on correct or real figures.

The figures listed in Table 5 are used to populate the Environment Specific Distributions (ESDs) (See Chapter 8) in our model. An example of how the input data relates to the *patient treatment process* is illustrated in Figure 5.

6.2.2 Security related elements

In the scenario, we will identify the processes related to the security measures of smart cards and passwords. Furthermore, we will identify security related KPIs and identify threats and vulnerabilities. In this thesis, *authentication failures* referrers to a authorized user who is unable to authenticate himself and in such is unable to gain access to a system. Our definition of a authentication failure does not include those failed attempts that does not result in a lock down of the user account, i.e. we do not include those events where a user misspells his password but is able to access the system by retyping the password correctly. A *authentication failure* by our definition results in a request to the ID office.

The two authentication mechanisms have been chosen due to the actual implementations done at Ahus. The effect each of the security measure has on the hospital will be determined by analyzing the affect each security measure has on the KPIs identified later in this Section and the *time relevant KPI* identified in the former Section. Each of the implementations have the potential of affecting the corner stone's of information security, namely confidentiality, integrity and availability. For example, we can determine how each of the measures reduces the number of incidents or how they affect the availability of the medical records. Furthermore, since the security measures are included in the BAs which we have identified earlier, we should also be able to detect any potential influence the decision has on these processes.

There are of course many other security measures relevant, including but not limited to, user training, backup and recovery, incident handling, and logging. Although we do not include such measures in our model, we recommend in Chapter 12 that such measures are attempted to be modeled based on the methodology in order to further test the simulations suitability.

Authentication mechanism processes

Although Ahus currently combines the use of both password and smart card mechanisms[8], we will assume that a homogeneous system is desired and will therefore determine the difference between the two security measures.

Each of the two authentication mechanisms have their own set of processes, as identified in the flow charts listed in Appendix H. However, the two authentication mechanisms share the same essential steps, as illustrated in Figure 6. When smart cards are used, the security guard stationed at the ID office, is, in collaboration with the human resources (HR) department and the IT department, responsible for enrolling the smart cards of new employees (step 2) and terminating the smart cards when the employees leaves (step 4). If passwords are used, these steps (2 and 4) are performed by the IT department in collaboration with the HR department. Regardless of the type of authentication measure used, the security guard is responsible for resetting forgotten passwords and PIN codes, and issuing smart cards (step 3). The security guard can either issue temporary cards or new cards depending on whether the employee only

[8]Smart cards are used to authenticate the user when accessing the operating system while passwords are used to authenticate the user when accessing DIPS and other similar applications

have forgotten the smart card at home, or if the smart card is lost, stolen or destroyed. In the case of passwords being used, the doctor will phone the ID office and the security guard will send a new password to a predefined cell phone number. When smart cards are used, the doctor must personally visit the ID office in order to reset the PIN code or get a new card.

Figure 6: Illustration of the authentication process

To simplify the scenario, we will only focus on step 3 in the *Authentication process*, namely the *Authentication mechanism maintenance* step. Within this step, three main activities are identified, as illustrated in Figure 7. Although this simplification eliminates both the enrollment and termination steps, the simplifications affect both security measures equally. In order to further simplify the model we will make an assumption with regards to the availability of the ID office and the employees that work there, namely that the delay of a request by the doctor will have the same triangular distribution regardless of when the request arrives. Normally, one would of course experience that outside office hours[9] the time delay before a requests by a doctor will take longer to complete since the security officer might not be available at the ID office, e.g. because he/she is conducting a routine control of the premises. However, since the simplification affect the two security measures equally, it should not affect the outcome. Another simplification which we make is that we do not include situations where the *ID office workers* become insiders, i.e. they deliberately issues smart cards or resets PIN codes and passwords to unauthorized persons. This simplification should affect both security measures equally.

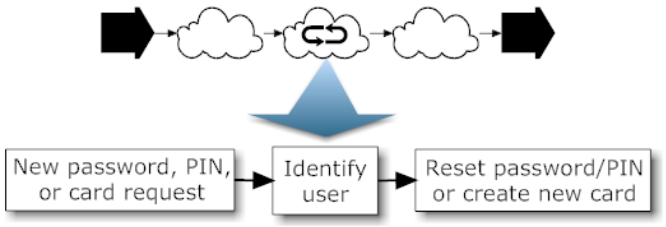

Figure 7: Illustration of the authentication maintenance process

Security related KPIs

Four different KPIs have been identified based on the standard for information security utilized by hospitals in Norway [54] and with conversations with the Chief Information Security Officer (CISO) at Ahus, Ellef Mørk. The KPIs are all linked to deviations identified based on the

[9] 07:30 AM and 16:00 PM according to Halvor Sandodden

Table 6: Security Related KPIs identified

Identified KPIs	Description	Source
Breach of medical record handling rules	The KPI focuses on the acceptance criteria of confidentiality. There is a zero tolerance with regards to such deviations.	Ellef Mørk
Lack of medical record integrity	The KPI focuses on the acceptance criteria of integrity and quality. Such a deviation can be caused by several reasons, both technical and human. A example of a requirement is that the loss of registered information in a medical record must not occur more than once each month.	Ellef Mørk and [57]
Medical record unavailable	The KPI focuses on the acceptance criteria of availability and integrity. The KPI is affected e.g. by the unavailability of a system, or when a authentication failure occurs.	Ellef Mørk, [54] and [56]
Use of emergency access	Since the use of emergency access grants a person access to more information than he/she normally should have access to, additional monitoring is needed. Because of this, the use of emergency access is classified as a deviation.	[54]

acceptance criteria's (in Norwegian: Akseptkriteriene) which hospitals in Norway define for confidentiality, integrity, availability and quality. The aforementioned standard is utilized in order to follow those laws and regulations relevant for the health sector in Norway.

Deviations, in this context, mean any handling of medical record related information that is not in accordance with existing regulations and procedures [58]. A deviation hence becomes similar to what we often referrer to as a incident in the area of information security. However, given that this is a health care specific scenario, we choose to mainly use the term deviation in this thesis. Examples of deviations are theft of equipment, employees not logging of system, employees sharing of password and user names with others, authorized users unable to access medical records, snooping, etc [58]. In addition to these deviations, the report to the management should also include unsuccessful log on attempts and authentication failures [53].

The identified KPIs which are listed in Table 6 is described in more details below.

Unlike the KPIs which were identified in the former Section, we are unfortunately not able to generalize these KPIs to the same level. The reason is that the effects on the deviation KPIs have different and conflicting causes, and there is not necessarily a connection between the deviation KPIs, although there might be a connection within some of the deviation KPIs[10].

Breach of medical record handling rules (In Norwegian: Pasientinformasjon på avveie)

This somewhat broad KPI focuses on the acceptance criteria of confidentiality. Several types of deviations are covered by this KPI, including breach of client/patient confidentiality, snooping[11], employees not logging of system, print out going to the wrong printer, employees sharing their password with others, and unauthorized access. Although this is not an exclusive list, it still illustrates some of the many possible causes for a deviation of this type.

Although there is a zero tolerance with regards to such deviations, this is still where the most

[10] E.g. a doctor that forgets his password, and hence experience a "medical record unavailable" deviation, might persuade another doctor into accessing the medical records on behalf of him, which by definition, would be a "Breach of medical record handling rules" deviation. We have however not included such interaction in the model.

[11] Non-service required journal access

deviations occur[12].

Lack of medical record integrity (In Norwegian: Helseinformasjon ikke oppdatert i journalene)
The KPI focuses on the acceptance criteria of integrity and quality, and can be caused by several reasons, both technical and human. An example of such reasons is that employees don't use the systems correctly, or that they don't know the rules, procedures and responsibilities. Such deviations could also be caused by technical/mechanical reasons, such as malicious code and disk crash to name a few.

There are many goals associated with this KPI, all depending on the specific deviation, but one example is that the loss of registered information in a medical record must not occur more than once each month [57].

Medical record unavailable (In Norwegian: Pasientinformasjon ikke tilgjengelig for behandlende personell)
The KPI, which focuses on the acceptance criteria of availability and integrity, will be affected by several causes, including unavailability of the electronic health record (EHR) systems (in Norwegian: Elektronisk Pasient Journal (EPJ) system), i.e. the system is down, or that the database that the EHR system utilizes is unavailable. Another reason for the occurrence of such a deviation would be that the user has forgotten his/her password or PIN code, or that the user has lost or forgotten the smart card.

Systems are prioritized based on the consequence of the unavailability [54]. For example, if the unavailability of a system becomes life-threatening for the patients or it becomes critical for the operations of the hospital, the system is classified as a priority one system. Should the unavailability result in considerable amounts of extra work or loss of effectiveness, the system is classified as a priority two system. Those systems which a prioritized system is dependent on, must receive the same level of priority [56].

Based on the prioritizing, the acceptance criteria must be determined. At Ahus, the acceptance criteria of unavailability for priority one systems are two hours[13]. An authentication failure, compared to the unavailability of EHR, is a isolated incidents that only affect one employee and his/her patient at a time. Although the duration of such incident are smaller than the aforementioned incident type, the frequency would be greater and the outcome are the same, namely that the medical record becomes unavailable for the doctor. It would however be wrong to state that the consequences are the same as the incident only affect one patient for a short time period. Although no specific goal is presented, it is desired to experience as few authentication failures as possible. This would both provide a better user experience and reduce the administrative costs[14]. However, since there are different procedures are used with regards to the resetting of passwords and PIN codes (and issuing new cards), one can not only determine how many authentication failures occur, but also relate this to the effects each failure has. The *time based KPI* can hence be utilized in order to determine the effects. In addition, factors such as additional hassle or other problems of a authentication failure must also be taken into consideration.

[12] According to conversations with Ellef Mørk
[13] Based on conversations with Ellef Mørk
[14] Although monetary costs are not included in the scenario we still draw the general conclusion that the monetary costs increase as the number of requests increase.

In addition to forgetting his/her password or PIN code, a employee could be exposed to an attack where a unauthorized user is able to reset the password or PIN code, basically resulting in a Denial-of-Service (DoS) attack as the employee no longer knows the password/PIN and hence don't get access to the system[15].

Use of emergency access (In Norwegian: Nødrettstilgang)

As one in general, and specifically in the health sector, experiences conflicts with regards of confidentiality and availability, one needs systems and methods for fulfill both of these demands in a satisfactory way. The use of emergency access is one such measure, but as this grants a person access to more information than he/she normally should have access to, one also needs additional monitoring. Because of this, the use of emergency access is classified as a deviation [54].

We will, in order to simplify the model, not include all incidents and deviations identified above. The KPIs that have been identified, both in this Section and in the former Section, can be divided into two categories, *Time relevant* and *Deviations relevant*. It should be noted that although the identified KPIs have been divided into two different categorized, the effects a particular security measures have on a business activity with regards of time delays is a direct result of the cross-category correlations that exists between the two categories. For example, if a doctor forgets his password and hence must contact the ID office ("*Medical record unavailable*" deviation occurs), this would affect the overall *time related KPI*[16]. How often such deviations occur and to what extend the deviation affects the business activity partially determines which security measures best suites the organization.

Identified security vulnerability, threats and consequences

A set of relevant vulnerabilities, threats and consequences of these attacks are listed in Table 7. This list includes only a small portion of possible threats, but since our intentions are not to perform a risk analysis, but rather determine the suitability of combining the simulation approach with the usage of KPIs, the identified threats should be sufficient for gaining this insight. To what extend each of the security measures mitigate the security threats, or at least minimize the success rate of them, will affect the KPIs and hence also determine which of the two security measures will best suited for the organization.

We can divide the identified threats into two different categories, where the first five threats are a result of what we define as *targeted attacks*, and where the last two threats have more natural causes. A *targeted attack* in this specific scenario is defined as an attack where the adversary must have physical access to the premises[17]. Such attacks can be performed both by outsiders and insiders, where insiders hence can be doctors. When the attacker is an insiders, the attack of interest is snooping [54]. Whether a *targeted attack* will occur highly dependents on the specific

[15] A more severe result of such an attack is of course that the unauthorized person gets access to confidential information. This does however require that the unauthorized persons either known the user name, which are not related to the users' real name [26], or has access to the smart card. Such an attack would hence also trigger the "Breach of medical record handling rules" KPI.

[16] Although not included, we could also see such a correlation the opposite way, e.g. if the average waiting time was high, we might see an increase in the number of deviations as one would work more effective to reduce the number of patients waiting, hence increasing the chances of deviations occurring.

[17] This is based on the demand described in [54] which states that *unauthorized persons* ***outside*** *the organization, regardless of resources and knowledge, shall not be able to access medical records*

Table 7: Identified security vulnerability, threats and consequence

	Vulnerability	Threats	Consequence
1.	Insecure storage of mobile phone and insufficient identification procedures when resetting forgotten passwords	Theft of mobile phone and attacker resetting password	Full unauthorized access to health information and the doctor in question is unable to treat patient, resulting in increased treatment time and reduced treatment quality
2.	User shares password with others	Attacker fools user into revealing password	Full unauthorized access to health information
3.	Insecure storage of smart card and insufficient identification procedures when resetting forgotten PIN code	Theft of smart card and attacker resets PIN	Full unauthorized access to health information and the doctor in question is unable to treat patient, resulting in increased treatment time and reduced treatment quality
4.	Insecure storage of smart card and user shares PIN code with others	Theft of smart card and attacker fools user into revealing PIN code	Full unauthorized access to health information and the doctor in question is unable to treat patient, resulting in increased treatment time and reduced treatment quality
5.	Medical records are available for all employees	Users perform non-service required journal access (snooping)	Employees get access to information which they are not allowed to access.
6.	Users forgets their password / PIN code	User unable to access the system	Unable to treat patient, treatment time increases, treatment quality decrease
7.	Users forgets their smart card	User unable to access the system	Unable to treat patient, treatment time increases, treatment quality decrease

situations within the hospital, e.g. that prominent people like the prime minister is admitted. Of course, an attack could occur regardless of such situations, but the probability of an attack to occur is always based on the motivation and resources of the adversary. Furthermore, as we identified in Table 7, the success of an attack often depends on several elements, but they are all based on the how successful the social engineering attack is. These assumptions suggest that the frequency of attacks will be low, but we suspect that should an attack occur, then the success rate will be high since a targeted attack most likely will be adapted to the specific attack. Should unauthorized access by outsiders occur, the results will also have the greatest overall impact, i.e. the consequence is catastrophic.

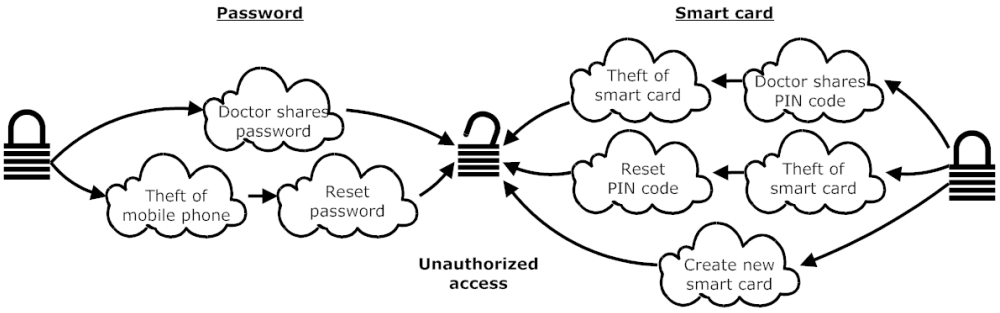

Figure 8: Illustration of possible unauthorized access attack paths

The possible attack paths an attacker can chose to gain unauthorized access is illustrated in Figure 8. Notice that we do not include brute force attacks. The reason for this is that such attacks will not be possible to conduct since the user account will be locked down or the smart card will be blocked if several authentication failures occur.

The complete interaction between all the threats, as well as the prerequisites for a successful targeted attack, is illustrated in Figure 40 found in Appendix I.

Input data collected

We will in this final Section present the security related input data which will be used in the model. The original data, as mentioned in the former Section, can be found in Appendix G.

The input data was gathered from several sources based on the scenario, the identified KPIs and security threats described earlier. Our main sources for this input data have been Ahus and Ergo Group[18]. In addition, due to lack of documented data, input data have in this Section in a larger degree been collected from surveys and other similar sources, as well as estimations. It is important to note that all figures presented in this Section are adjusted based on the scenario, i.e. the figures are based on 12 employees. The "per-user" figures, which the input data presented in this Section is based on, are found in Appendix G. The input data found in Table 8 consist of the Security Related Distributions (SRDs), which is used to populate the *triangular distributions* in our model. More information about the SRDs and the triangular distributions are described in Chapter 8 and Chapter E, respectively.

Table 8: Input data - Security Related Distributions

Parameter	**Time Between Arrivals (TBA) / Duration**			**Source**[1]
	Minimum	**Maximum**	**Most likely**	
TBA of password authentication failure	29589 min	32727 min	31533 min	D
TBA of PIN code authentication failure	67081 min	167442 min	101408 min	D
TBA of replacement card authentication failure	220408 min	400000 min	295890 min	D
TBA of temporarily card authentication failure	121348 min	284211 min	191150 min	D
New password duration	3 min	10 min	5 min	SME
New PIN duration	6 min	17 min	11 min	SME
New PIN duration @ PAC	1 min	3 min	1 min	SME/E
New card duration	10 min	21 min	15 min	SME
New card duration @ PAC	3 min	5 min	4 min	SME/E
TBA of snooping	21600000 min	32400000 min	25920000 min	SME
TBA of unauthorized access attempt	103680 min	518400 min	259200 min	E

[1] Sources are either Documented (D), Subject Matter Expert (SME), or Estimated (E)

Recalling the description of the identified security threats in Section 6.2.2, we can divide the parameters found in Table 8 into two categories, namely the occurrence of targeted attacks and the occurrence of natural caused threats, where the recover duration from the latter threats are also included.

The input data used for the occurrence of natural caused threats, e.g. a doctor forgetting his password or losses his smart card, are all based on documented data. The first set of input data, "Password authentication failure", is based on data received from Ergo Group, while the remaining three sets of input data are based on data from Ahus' ID office. As we see from these figures,

[18] Ergo Group is a leading Nordic IT company and was used due to lack of access to password related data from Ahus.

the occurrence of forgetting passwords are more frequent than smart card related occurrences, where, in general, for each 31522 minutes (22 days) one of the twelve doctors will forget his password, while only for each 196149 minutes[19] (4,5 months) one of the twelve doctors forgets his PIN code or needs a new card. The fact that this specific input data had to be collected from different source is unfortunate as we cannot compare the original data directly. However, since this data, as with the rest of the input data, is processed equally, the problem is minimized. More information about the original data and the processing of this can be found in Appendix G.

The input data for recovery duration, i.e. the time it takes from a doctor experience an authentication failure until he again is able to authenticate himself, are based on the SME opinion of Ellef Mørk and Halvor Sandodden at Ahus. When a doctor forgets his password, it takes between 3 to 10 minutes, where 5 minutes are considered to be most likely, from when a doctor calls the ID office and request that a new password is sent to his mobile phone, until he has received the password. Because doctors who forgets his PIN code or needs to a new, or temporally, card, needs to physically visit the ID office in order to reset the PIN code or receive a new card, the recovery duration is longer than with passwords. The added duration includes the time it takes to walk to the ID office from wherever the doctor is located at the time of authentication failure, and back in order to continue the work. Since it takes longer time to issue a new card than to resetting the PIN code, the most likely recovery duration, including the walking time, is 15 and 11 minutes respectively. An example of how the input data relates to the *authentication maintenance process* is illustrated in Figure 9 where the password related input data is utilized.

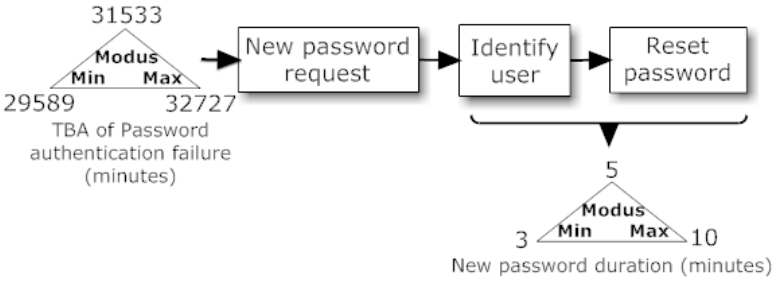

Figure 9: Illustration of the password maintenance process with input data

Regardless of which security measure is selected, all doctors must use their ID card to access the premises when arriving at work. Because this physical access control (PAC) is closer to the ID office, the recovery duration are quite low. In the case of smart cards being used as a security measure, we will assume that the ID card which is used in fact is the smart card, and hence, the same PIN code is used. When passwords are used, we assume that the number of authentication failures with respect to the ID card is similar with the smart cards.

Finally, we have two types of *targeted attacks*, namely snooping and unauthorized access attempts. Although snooping by definition also is unauthorized access, we chose to distinguish between snooping and unauthorized access attempts which is performed by non-employees.

[19] (101408 + 295890 + 191150) / 3

While the snooping figures are based on SME opinions, we have not been able to access any data in which we can justify the number of unauthorized access attempts at the same level. Recall that we in Section 6.2.2 stated that such attacks will be highly dependent on the specific situations which occur within the hospital. Because of this, it becomes difficult to determine such figures with any confidence. However, we assume a "worst case" scenario and apply data from other sources (see Appendix G). Although the figures them self most likely are "wrong", they affect the two security measures equally, and should therefore not affect the decision with regards to which of the two security measures are best. For more information about the figures utilized, see Appendix G.

As with the number of "unauthorized access attempt", the success rates of such attacks have also been difficult to determine. The success rates, listed in Table 9, have been estimated by the author, although some of the figures are based on data found on similar situations. With the exception of snooping, there exists much uncertainty in the figures presented in Table 8. Furthermore, unlike the other figures, the estimations made affect the two security measures differently. Because of this, the selection of success rate can potentially have a large influence on the final outcome with regards to the security measures applicability to the organization. We must hence take this into consideration when analyzing the results later on.

Table 9: Input data - Success rate of targeted attacks

Attack type	**Success Rate**	**Source**[1]
Snooping	100%	SME/E/D
Get password/PIN	50%	SME/E/D
Reset password	50%	E
Reset PIN	20%	E
Get new card	10%	E
Theft of mobile	2%	E/D
Theft of card	2%	E

[1] Sources are either Documented (D), Subject Matter Expert (SME), or Estimated (E)

We assume that the number of snooping attempts will be the same for each type of security measures[20], and that should a doctor chooses to conduct such unauthorized access he/she will also succeed. The three parameters, "Get password/PIN", "Reset password" and "Reset PIN", are all based on social engineering attacks, and recalling that we assume that the attacks are targeted, the success rate also becomes relatively high.

Figure 10 includes the frequency and the success rate figures identified for the possible attack paths which was identified earlier in this Chapter.

How successful an attack is depends on several elements, and an attack can also be accomplished in several different ways. When passwords are used as the security measure, the attacker can either gain access by receiving the password directly from a doctor, or by receiving a new password from the ID office. The latter method requires that the attacker also has the doctor's mobile phone as the password is only sent to the number which is registered by the ID office. In both these cases, the attacker utilizes social engineering as a attack method, and will also need to steal the mobile phone in the latter case. When smart cards are used as the security measure,

[20] See Appendix G for more details

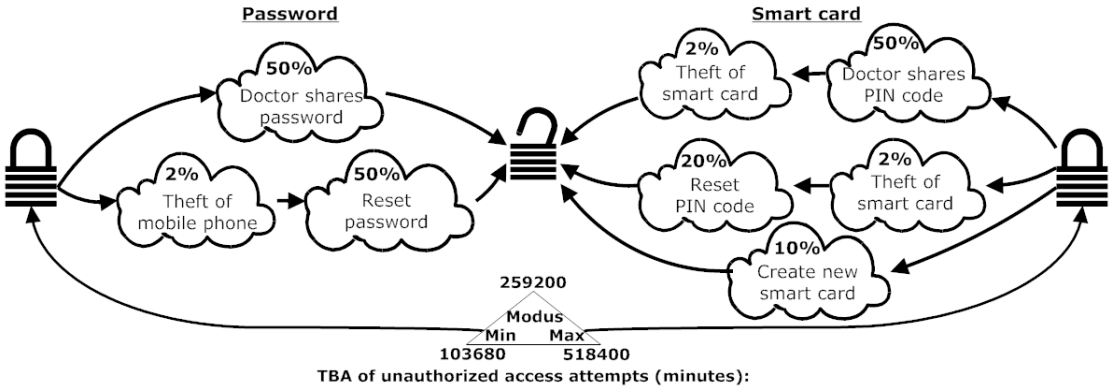

Figure 10: Illustration of possible unauthorized access attack paths with success rates and frequency

we simulate that the attacker first of all needs the smart card, which either must be stolen from the doctor or issued by the ID office, and secondly, that the attacker needs the PIN code, which again either must be gained from the doctor (similar to getting the password) or by resetting it at the ID office. The prerequisites a attacker needs in order to perform a successful attack when smart cards are utilized, suggest that the overall success rate becomes less. Recall that we do not assume that *ID office workers* become insiders. However, if we had includes such situations, the success rate of attacks depending on bypassing the ID office would naturally increase.

6.3 Discussion

In the methodology described in Chapter 4, as well as in the methodologies described in [46] and [35], data collection is identified as a important step in the success of a modeling and simulation process. Because of the importance of data collection, we have attempted to gain as much of the needed data directly from our collaborators. We have also utilized the framework described by Olsen ([61]) in order to help us identify objects and elements in the scenario which an adversary might exploit. Furthermore, other sources have been utilized in order to gain access to additional data where needed. However, by utilizing several different sources, where some of the data sets are quite different, issues will emerge. In order to compensate for different populations in the data sets, all data have been processed in order to get a uniformly scaled value such that the data could be reused in our scenario. However, a larger issue is related to the validity of some of the data used. Although we have had a focus on collecting data from area specific sources, e.g. Ahus and the Rheumatism Hospital, we have unfortunately not been able to collect all the needed data from such sources. Because of this, there is much uncertainty surrounding the validity of the number of *unauthorized access attempts* and the success rate of the different types of attacks. Since the success rates affect the two security measures differently, this will also affect the validity of the results from which the decision is to be made. Should the data which is utilized in the simulation not reflect the real world, the direct results are that the wrong security measure will be selected, i.e. a decision is made based on wrong assumptions. The importance of the validation and verification phase has therefore also been identified as very

important according to the methodologies described in [4], [46] and [35]. However, since our objective has never been to perform an actual assessment of the implementation of the security measures at Ahus, we have not focused on verifying the data or validating the model. However, such simplifications must then also be taken into consideration when analyzing the results of the case study. Although we don't include these rather time consuming steps, we believe that we are still able to determining the suitability of the simulation approach.

Because of the importance of collecting good data, simulation and modeling are also depended on the organizations current level of maturity with regards to information security. For instance, if the organization already has performed a risk analyzes the results from this will be of great help in order to simplify the input data process. Furthermore, if the organization already has a set of metrics which they have identified, this further helps to determine which information to utilize. An important issue to note is that we have gained access to information based on security measures which already where implemented. Since one of the objectives with the simulation approach is to determine what the results of a planned security measure will be, such information might not be available. Hence, it becomes even more difficult to collect the needed data.

During the construction of the scenario, several simplifications have been made which affects both the scope of the scenario, and the collected data. For example, we have only focused on doctors and only on a small set of business activities. The justification of these, and similar, simplifications are that we believe that we are able to determining the suitability of the simulation approach without this additional complexity to the scenario. However, although we don't include the additional complexity in the scenario, when determining the suitability of the simulation approach, we need to take into consideration the fact that such elements are in fact needed to be included when creating a model of a real system. Another example of a simplification made with similar justifications are made with regards to the time based KPIs identified. We have only considered one single time related KPI, instead of the four time based KPIs which we in fact identified.

Recall that although KPIs can be both financial and non-financial measures [80], Jaquith ([40]) argued that both time and money should be incorporate. In our specific scenario, costs have only been included in the sense of determining how the security measure affects the time used on a business process. Although we do not utilize monetary costs in our scenario, we did identify several who included such costs in their work, including Bartolini et al. ([5]), O'Gorman ([60]), Helkala ([26]) and Aiber et al. ([1]). The fact that we have not included such KPIs in our scenario can hence be determined as a limitation. However, a couple of arguments support our selection of KPIs. First of all, the health sector and most organizations within the private sector have a somewhat different focus with regards to KPIs. Whereas the health sector have a focus on quality with regards to time related KPIs, e.g. waiting time etc., most private organization have a greater focus on profit. We have also seen a similar non-monetary focus by Holm et al. ([29]).

In addition to the *time related KPI*, we have also identified several deviation based KPIs. In [57] the consequences of the occurrences of some deviations have been determined as catastrophic. Furthermore, considering that the economical model presented by Gordon et al. ([23]) does not cover protection of assets or other circumstances where a loss could be catastrophic,

we see that monetary cost might not always be relevant. In such, when determining the effects a security measure has on the hospital, deviations, and not money, becomes the important measure of performance.

However, monetary costs will in other scenarios be a relevant and important factor. It is hence important to note that both the methodology in specific, and the simulation approach in general, are able to consider the factor of monetary cost in the analysis of the security measures. This was for example done by Cohen in [12] where per-use cost and fixed costs associated with computer attacks are considered with the use of simulation. The combination of monetary and non-monetary costs have also been identified. For example, O'Gorman ([60]) defines a strong system as where the cost of attack is greater than the potential gain to the attacker, and where the cost includes money, time used and the potential for criminal punishment. The fact that monetary costs in this specific scenario have not been included should therefore not in itself be a factor which affects the considerations with regards to the suitability of the simulation approach.

Since we have not included monetary costs in this scenario, issues surrounding administrative costs of each security measure have not been included. By excluding the turn-over of employees, as well as the enrollment and termination steps identified in Figure 6, the administrative costs associated with these processes are not included. However, should monetary costs have been included, the cost of enrollment and termination would also have be needed to include. Furthermore, since the certificates on each smart cards expires after three years, on would need to include such additional costs.

7 Case Study

In this Chapter we present a case study which, based on the methodology described in Chapter 4 and the scenario described in Chapter 6, analyses the effects of implementing smart cards as opposed to using passwords in a hospital. As described in Chapter 6, we have simplified the scenario considerably. These simplifications have been possible to carry out as our goal is only to determine the suitability of the simulation approach, and not conduct an actual assessment of the true affects of the security measures on Ahus. Based on this objective, these simplifications and limitations should not affect our final conclusions.

7.1 Preface

The case study has been conducted in a iterative fashion, where several realizations have been made with regards to the first step in the methodology described in Chapter 4. The main idea is to reduce the complexity of the model construction phase by not including the temporal relationship between the different business activities. That is, we do not model the business process directly. Instead, we identify events, objects and other elements that are relevant for the security decision, and determine how a object responds to a specific event. In other words, we create a model based on the state transitions. These realizations have also resulted in a new methodology, called Minimalistic Model Design (MIMD) methodology, which we will describe in Chapter 8.

7.2 Case study: Determining the effects of implementing smart cards for authentication in a health care environment

In this Section we will first identify and describe the KPIs, objects, and events relevant for the case study. We then determine the mapping between these objects and events, and determine the objects responses to specific events. The created model is then briefly described, before the simulation parameters are estimated. Finally, an analysis of the simulation output is conducted in order to determine the effects of each security measure and ultimately selecting the "best" security measure.

7.2.1 Design and data collection

The first step in the data collection phase is to identify the KPIs and their underlying events. Based on the Scenario described in Chapter 6, the KPIs have been identified, and are illustrated in Figure 11. As we can see in Figure 11, relevant events have also been included.

After identifying the KPIs, we need to identify relevant objects. Based on the scenario, a total of eight objects were identified, as illustrated in the object tree (Figure 12).

Based on the scenario, in particular the process illustrated in Figure 3 and 6 in Chapter 6, we have identified relevant *events*. The events, listed in Figure 13 and described below, are divided into three main groups, namely "Business activities", "Security measure events" and "Security threats and vulnerability events".

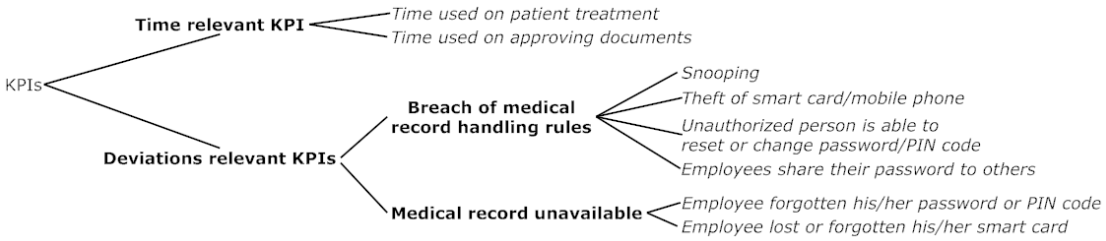

Figure 11: Identified KPIs for case study

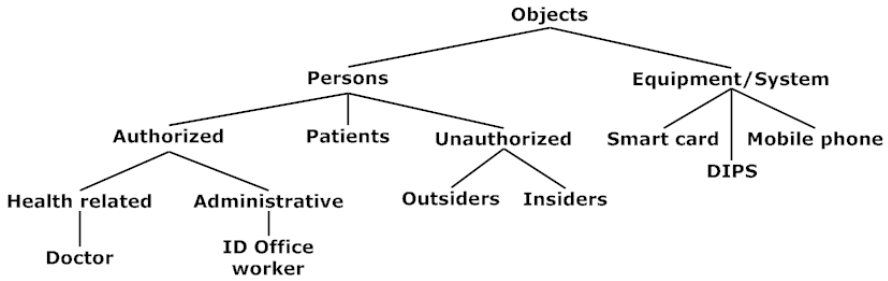

Figure 12: Identified objects for case study

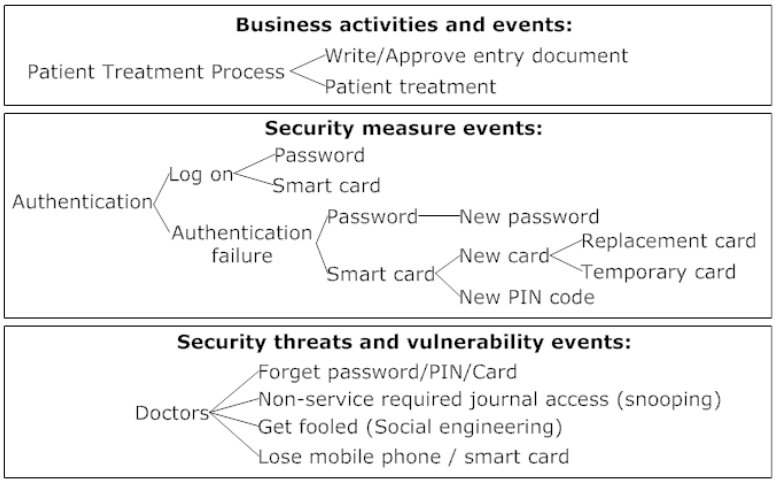

Figure 13: Identified events for case study

Based on the scenario, we find in the first main category the business activity (BA) "*Patient Treatment Process*". Within this BA, the events of "*Approve documents*" and "*Patient treatment*" are identified. The second category, the "Security measure events", we again find only one relevant event, namely "Authentication". The activity includes both the authentication process itself as well as the process of resetting and replacing the authentication mechanisms. Recall that we do not include failed attempts that do not result in a lock down of the user account. The events are further divided based on the different security measures of interest, namely "Password" and "Smart card". Each of these mechanisms includes the event of logging on to DIPS and the event of resetting the password or PIN code. In the case of "Smart Card", the event of replacing the card, either with a temporary replacement card or with a new card, is also included. The final category consists of security threats and vulnerability relevant events for the object "Doctor".

The mapping between the identified KPIs, objects and events are illustrated in Figure 14, where the squares, circles and clouds represents the KPIs, objects and events, respectively. The green lines represents mapping that are considered as positive or normal interaction, red lines represents negative interaction. We have further chosen to represent the interaction between the "patient", the "patient treatment" and the "time relevant KPI" with black lines as the interaction, whether positive or negative, depends on the rest of the interactions. The gray stippled line differentiates between the authorized and unauthorized objects, where doctors are found in between as they can become insiders[1]

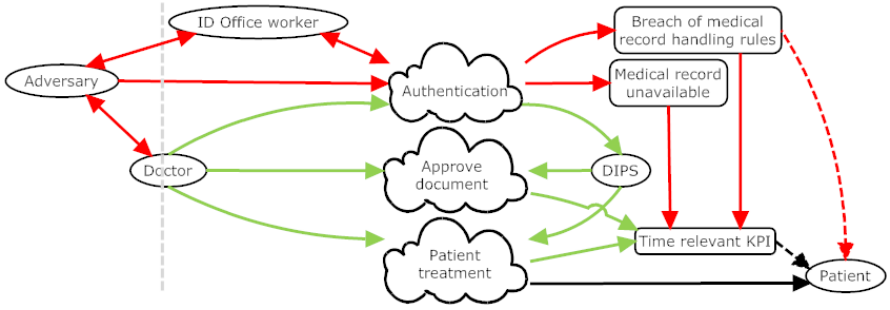

Figure 14: Mapping between identified KPIs, objects and events

Finally, from a doctors "point of view", the possible states (response to events) have been identified, as illustrated in Figure 40 (Appendix I). In Figure 40, "*Others*" represents both *outsiders* and *insiders*, who could gain access given a set of prerequisites. The illustration does not take into account the different probabilities for each of the state transitions, but focuses only on those elements that could occur.

7.2.2 Building the model

We will in this Section describe the model which have been built based on the information identified in the former step, and on the input data described Chapter 6. The model is based on a state/action concept, where the states of the objects are determined based on the activity of

[1] ID office workers could also become insiders, but this have not been included in the scenario.

the rest of the model.

We have chosen to not describe the model in details this Section as this will become too simulation environment specific, and hence remove focus from what is of importance. We will however, due to the centrality of the "*patient treatment process*" in the case study, describe the model structure of this process. For illustration purposes, Figure 15 depicts how we have constructed this specific process in the model. The notations used are explained in Appendix D.

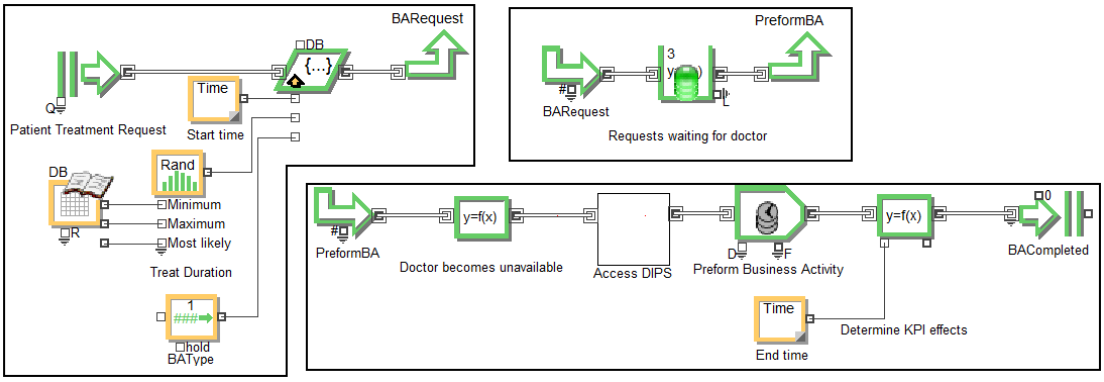

Figure 15: The model structure of the patient treatment process

The patient treatment process, as with the model in general, is separated into smaller individual modules. The reason for such separation lies in that although all employees will have specific business activities, they will all share a common denominator, namely that they perform a Business Activity (BA). Such separation hence provides us with a structure that allows scalability and re-usability[2], by allowing new objects and new business activities to easily be added without affecting the overall model structure. In addition, we are able to monitor all activities in one central point in the model which both saves us time and reduces the chance of errors occurring. Although new requests and their properties are randomly generated within a simulation run, the implementation of Common Random Numbers (CRN) (see Chapter 4 for more details) will yields the same number of requests with the same properties when comparing the security measures. That is, a request will be generated at the same point in time for each simulation run, and the *i*th request will have the same properties in each simulation run. When a request is generated, it is placed in a queue awaiting a available doctor. When a available doctor arrives, it is processed by the "*Perform Business Activity*" "block" based on the properties of the request, which was determined at creation of the request. When a request is processed by the doctor, he or she must access DIPS in order to start performing the business activity. If the employee is unable to gain access to DIPS due to authentication problems, i.e. forgotten password, PIN code or lost smart card, the doctor will need to contact the ID office.

As we have seen so far, the structure of the model consists of several different "blocks". However, the true work is conducted with the utilization of the underlying database. The database

[2]The model allows other types of scenarios to be modeled with little additional work needed.

is an important element, and is used to set initial starting values and access the different statistical distributions, as well as managing each person's states (e.g. whether a doctor remembers his/her password) during the simulation run. By utilizing the database for such tasks, we are able to access the same data from different places within the model, and managing the data becomes much easier. Which values to use, and which attributes/fields to include, are amongst other things based on the input data (see Chapter 6 and Figure 40). The values in the database are altered based on logical functions found inside specific types of "blocks". However, each simulation run is initialized with the same set of *start values*. By utilizing the same initialization values for each of the two simulation runs, one for each security measure, we achieve the best possible output with regards to comparing the two security measures.

We will not go into more details about the interaction found within the model in the report. However, the variables utilized in the database and the state transitions which occur in the model are described in Appendix I.

7.2.3 Simulation parameter estimation

In this Section we will determine the simulation setup for both types of security measures, i.e. determine the warm up period and apply the method of batch means to eliminate the effects of autocorrelation (Appendix E).

Selecting random number streams

Based on the CRN implementation guides described in Chapter 4, different seeds have been used in each block in the model which generates random numbers. This way, all blocks generate their own set of random numbers within a simulation run, but since the same seed is used for each simulation run, the two security measures are compared based on the same random numbers.

Measure of performance

In our case study, several measure of performance is of interest. The first measure of performance is the mean response time of a business activity per request, where a response time is defined as the total time from a request arrives until it departure from the system. An observation hence becomes the time used on a specific request and this answers our question whether a particular security measure affects the time related KPI. In addition, we want to determine the number of deviations observed with each type of security measures. We have here several observations of interest, where the first is authentication failure when accessing DIPS, i.e. the number of authentication failures are one observation. We are also concerned with how many snooping events occur as well as the number of attempts of unauthorized access. We differentiate between successful and unsuccessful unauthorized access attacks, as this will help us to determine the effectiveness of the security measure.

The performance measures are to be determined with a 95% confidence interval ($\alpha = 0{,}05$).

Determining the number of replications

The number of replications, as described in Chapter 4, is determined based on whether the system is terminating or non-terminating. The system described in our scenario is of a non-terminating type given that there is activity 24 hours each day and since there are not a specific time where no more relevant data is generated. Because of this, one single replication is chosen,

instead of several replications.

Determining the simulation length

As described in Chapter 4, there are several steps involved in determining the simulation length. First, we determine the warm up period in order to remove sufficiently amounts of biased data. Then, based on the remaining data, we apply the method of batch means to determine the number of batches needed and the size of each of these batches. We then determine if this simulation length produce output that is within our confidence interval. Should this not be the case, the simulation length will be increased.

Warm up period

Determining the length of the warm up period is important since the individual observations in the beginning of the simulation run will be influenced by the start up effect of the simulation, e.g. all employees will be available in the beginning of the simulation run. Based on the methodology, we determine the warm up period by visually confirming when the cumulative average of the individual observations stabilizes. Each run outputted two graphs, one for each of the business activities, and each graph includes all individual observation, i.e. the time a doctor uses on a specific business activity request and the cumulative average.

The individual observations and the cumulative average identified for the business process "Patient Treatment", when the security measure "Password" was selected, is depicted in Figure 16[3]. The remaining three graphs can be seen in Figure 53, 54 and 55 found in Appendix K.

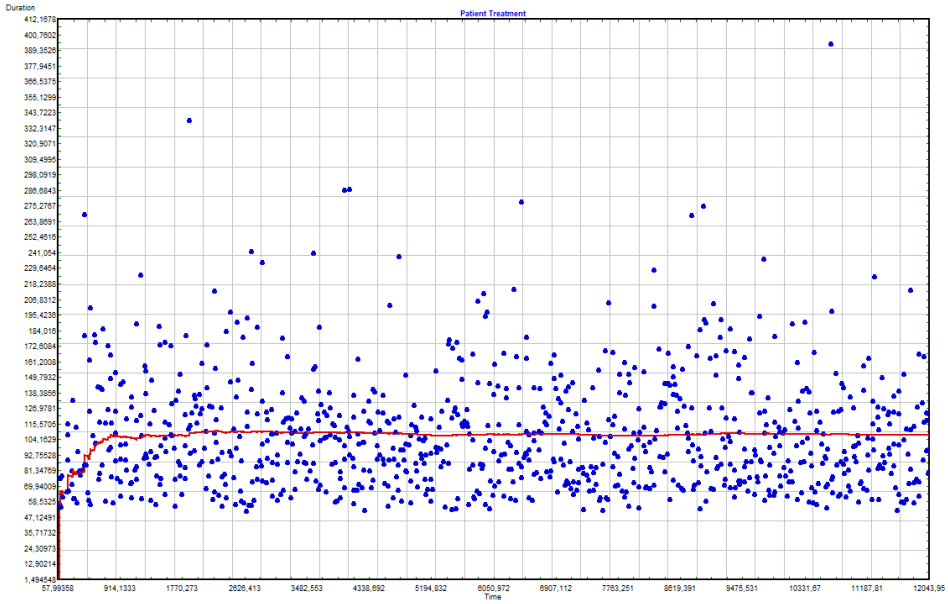

Figure 16: Individual observations and cumulative averages for Patient Treatment with password

[3]In all of the Figures, the x axis represents the simulation time while the y axis is the duration of each observation. A observation is either the time used on a "patient treatment request" or a "approve document request"

The warm up period (WT) for the business activity "Patient Treatment" is about 2500 minutes for both types of security measures. For the business activity "Approve Documents", the warm up period ranges from about 60000 minutes to 100000 minutes. In order to later compare the results, we choose to use the highest warm up period for each type of business activity, i.e. all observations between $T = 0$ to $T_0 = 2500$ minutes for "Patient Treatment" and between $T = 0$ to $T_0 = 100000$ minutes for "Approve Documents" will be deleted and not used in the next step.

Creating batches

In order to process the output and ultimately determine which of the two security decisions are the "best", the output from the two simulation runs are batched according to the process described by the methodology. This process of creating the batches is summarized in Table 10 where the number of observations, before, during and after deletion are listed. The process is described in more details below.

Table 10: Number of observations gathered, deleted and used

Simulation run (SimRun)	**SimRun1 (Password)**		**SimRun2 (Smart card)**	
Business Activity	Patient Treatment	Approve Documents	Patient Treatment	Approve Documents
Observations collected	388764	70694	388763	70694
Deletion due to warm up period	208	2500	208	2500
Deletion due to batch adjustment	56	114	55	114
Remaining observations	388500	68080	388500	68080
Number of batches	370	370	370	370
Batch size	1050	184	1050	184

Based on the identified warm up period of 2500 minutes, we need to delete the first 208 observations gathered from the "Patient Treatment" activity. This particular number of observations is based on the minimum time between arrivals (TBA) of the "Patient Treatment" requests of 12 requests per minute, suggesting that at the most, 208 observations will be observed during 2500 minutes. Likewise, we will delete the first 2500 observations gathered from the "Approve Documents" activity, based on the WT of 100000 and on the minimum TBA of 40 requests per minute. Although we observed a shorter warm up period when the smart card measure was selected, we still delete the same amount of observations for reasons already mentioned.

In total, 388764 and 388763 "Patient Treatment" observations was collected when the security measures password and smart card was selected, respectively. In the case of "Approve Documents", a total of 70694 observations were collected with both types of security measures. As we can see, after deletion, we still fulfill the recommendations found in Chapter 4 with regards to collecting 10 times as many observations as is deleted.

The remaining observations needs to be divide into as many batches as possible, without removing a unnecessary amount of observations. According to the recommendations in Chapter 4, around 400 batches should ideally be created. However, the highest common number of batches which saved the greatest number of individual observations in combination with easily allowing us to rebatch the observations later on was 370 as we will describe later.

In order to batch the observations into 370 batches, we need to reduce the number of "Pa-

tient Treatment" observations to 388500 and the number of "Approve Document" observations to 68080. We choose to remove those observations that are collected the earliest in the simulation run as these are considered to be the observation that are the most affected from the warm up period, although this should be minimal given the deletion phase. This process results in 370 batches with a batch size of 1050 and 184 for the "Patient Treatment" and "Approve Document" activities, respectively. Following the methodology, the next step is to utilize Equation 4.1 in order to estimate the sample lag-1 autocorrelation[4] based on the means of each batch. The results from this calculation are listed in Table 11.

Table 11: Results from estimating the sample lag-1 autocorrelation ($\widehat{\rho_1}$)

Simulation run	**SimRun1 (Password)**		**SimRun2 (Smart card)**	
Business Activity	Patient Treatment	Approve Documents	Patient Treatment	Approve Documents
Estimated autocorrelation ($\widehat{\rho_1}$)	-0,136	-0,003	0,043	0,054

Since the sample lag-1 autocorrelation of the batch means, $\widehat{\rho_1}$, are within the recommended threshold[5], we rebatch the observations into 37[6] batches. Based on the new batches we calculate the standard deviation of the point estimate and from this, calculate the 95% confidence interval. The results from these calculations are found in Table 12.

Table 12: Calculating sample mean, variance and the 95% confidence interval

Simulation run	**SimRun1 (Password)**		**SimRun2 (Smart card)**	
Business Activity	Patient Treatment	Approve Documents	Patient Treatment	Approve Documents
Sample mean ($\overline{Y}$)	106,23	99,49	106,30	97,92
Standard deviation (σ)	0,5	2,38	0,61	2,77
Half-length CI (H)	0,16	0,77	0,2	0,89
Confidence interval (CI)	(106,07;106,39)	(97,72;99,25)	(106,11;106,50)	(97,03;98,82)

As a final step in the methodology, in order to determine the independence of the batch means, we compute the test statistic from the 37 batches based on Equation 4.3. That is, we need to determine if $C < z_\beta$. Recall from Chapter 4 that if all C's are less than 1.96 (z_β)[7], we have confirmed the independence of the batch means and we are then ready to compare the systems. As we can seen in Table 13, all the test statistics are indeed less than 1.96, and we can therefore continue to analyze the results from the simulation runs.

[4] See Appendix E for a description of why we need to determine the autocorrelation.

[5] The recommended threshold according to the methodology described in Chapter 4 is less than 0.2

[6] We increase the batch size by a factor of 10, which provides us with a number of batches close to the recommended number of batches (Chapter 4). The simple transition from 370 to 37 is also why we selected a number of batches earlier which, when divided, yield a integer.

[7] The value is based on the Cumulative Normal Distribution appendix found in [4]

Table 13: Results from computing the test statistics

Simulation run	**SimRun1 (Password)**		**SimRun1 (Smart card)**	
Business Activity	Patient Treatment	Approve Documents	Patient Treatment	Approve Documents
Independence of batch means (C)	-0,67	0,05	0,40	0,39

7.2.4 Analyzing the simulation output

One of the goals with the analysis is to determine the time difference a doctor experiences with a Business Activity (BA) when implementing the security measures. The other goal is to determine the number of deviations observed with each of the two security measures. That is, we are interested in determining which of the two authentication mechanisms provides the best overall security level while simultaneously not negatively effecting the business activities. Since we determined in the former Section that the length of the simulation run was sufficient, and since we have deleted the initial data, we can start analyzing the simulation output from each system configuration.

Recall from Chapter 6, that the security measure which yields the lowest *time related KPI* value, is considered the "best" security measure. Hence, we are interested in determining which of the two authentication mechanisms provide the lowest "time usage". In order to determine which security measure which yields the lowest *time related KPI* value, the difference between the observations from the simulation run are calculated. In specific, the difference is calculated by subtracting the observed "time usage" with smart cards (θ_2) from the observed "time usage" with passwords (θ_1), i.e. θ_1 - θ_2. By calculating the confidence interval from the observed difference we can then determine which of the two mechanisms is "best". Should the confidence interval only include negative numbers, the password scheme is determined best since all observations then indicate a lower "time usage". Likewise, should the confidence interval only include positive numbers, the smart card mechanism will be determined the "best" with regards to "time usage". The results from the calculation can be seen in Table 14 whereas the output which these results have been based on is listed in Table 30 (Appendix K).

Table 14: Determining the best security measure with regards to the *time based KPI*

	Difference (θ_1 - θ_2) between password (θ_1) and smart card (θ_2)	
Business Activity	**Patient Treatment**	**Approve Documents**
Sample mean ($\overline{Y}$)	-0,07 min	0,56 min
Standard deviation (σ)	0,72 min	3,39 min
Half-length CI (H)	0,24 min	1,11 min
Confidence interval (CI)	**(-0,31;0,17)**	**(-0,55;1,67)**

As we can see from the results listed in Table 14, the two confidence intervals in fact includes both negative and positive numbers, i.e. 0 (zero) is included in the confidence intervals. This result means one out of two things. Either, there is a difference between the two authentication mechanisms but we have not observed this, or there are in fact no statistical significant difference between the two security measures. The former issue would be a result of comparing data which is based on a too short simulation run. However, we where able to conclude in the former Section

Table 15: Determining the best security measure with regards to the *number of deviations*

KPI	**Authentication mechanism**	
	Password	**Smart Card**
Medical record unavailable		
Authentication failures PAC (% of total)	87 (0,02%)	41 (0,01%)
Authentication failures DIPS (% of total)	174 (0,04%)	46 (0,01%)
Breach of medical record handling rules		
Snooping	0	0
Unauthorized person gains access (Total number of attempts)	9 (16)	0 (16)

that the simulation run was in fact sufficiently long. Therefore, we can conclude that the result indicate that there is no statistical difference between the two security measures with regards to the added time a doctor spends on a business activity.

In order to determine which of the two security measures is best, we therefore instead turn to our other measure of performance for guidance. Table 15 lists the number of deviations registered during the simulation run with each type of security measure selected. In order to place a specific number into perspective, we also list how many percent a figure is with regards to the total number of DIPS accesses, or the total number of attempted unauthorized access attacks.

As opposed to the *time related KPI*, we observe a difference between the two security measures in both types of deviation related KPIs. Recall from Chapter 6, that different KPIs have different goals. In order to determine which security measure can be determined the "best", we therefore need to compare the results based on the goals. While there is a zero tolerance for any "*breach of medical record handling rules*", there is no such limit or threshold with the "*medical record unavailable*" KPI. However, recall that it was determined in Chapter 6 that it is desirable to have as few authentication failures as possible.

The number of authentication failures observed when a doctor accesses DIPS is almost four times higher when using passwords compared to using smart cards. With regards to the number of authentication failures at the physical access control (PAC), there is also a difference between the two security measures with regards to the number of failures, where in the case of passwords being used there is twice as many failures. There are however some issues with regards to these figures that we will discuss later. For now, we notice that there is a difference between the two security measures with regards to the total number of authentication failures, both against PAC and DIPS, in favor to the smart card.

As we see from Table 15, the number of snooping attempts was zero. The reason for this is simply that the simulation length was to short in order for such rare events to occur. Recall that it is assumed that the number of snooping attempts will be the same for each type of security measures, and the simulation length was therefore not adjusted accordingly, as this would not have yield a different results with regards to determining a difference between the two security measures.

The results which differentiate the two security measures the most, is with regards to *unauthorized person gains access*. When smart cards were selected as the security measure, none of the 16 attempts were successful, while 9 of the total 16 attempts was successful when passwords were used. As the consequence of such deviations in addition will have the greatest overall

impact, the results become important with regards to the choice of security measure. It is important to notice that these results are based on input data which are estimated or adapted based on statistics from other areas, and are not to the same degree as the other input data valid. Because of this, the results should be treated accordingly. Our intentions have however never been to determine what is the best choice of the two security measures, but rather determine if the simulation approach was suitable for make such decisions.

Discussion of the simulation output

As mentioned earlier during the analyses of the results, there are some issues that need to be discussed. First of all, although the occurrence of authentication failures are four time higher when passwords are used, the additional hassle[8] a doctor experience when forgetting his/her PIN code or smart card, is not taken into consideration. This additional hassle could become a motivation for not forgetting the PIN code or card again. More severely it could result in other types of deviations in addition to the authentication failure itself, namely that a doctor, instead of resetting his own PIN code right away, instead borrow another doctors smart card. Such a situation would as mention yields more problems, and is also not as easy to detect. Although we observed four times as many authentication failures when passwords are used we still concluded that there where no statistical difference between the two security measures with regards to the added time a doctor spends on a business activity. A logical explanation for this is the fact that it takes more time to reset a forgotten PIN code, or get a new smart card, than it takes to reset a password.

Since we assume that the physical access control is utilized with both types of security measures, an issue arises. In the case of smart cards, we assume that the same smart card and PIN code are used both to access DIPS and to enter the premises (PAC). This result in dividing the number of PIN code resetting and new cards issued between DIPS and PAC. However, in the case of passwords, we need to use two different distributions in the same simulation run, one for passwords and one for the ID cards. Because of this, all the occurrences of authentication failures will occur at the PAC, whereas in the case of smart cards, these occurrences were divided between the two. The question, or issue, then becomes, does a doctor who uses his card more often (both for accessing DIPS and PAC) reduce the number of authentication failures compared to only using the ID card at PAC, or will the added number of times the smart card is used (accessing DIPS 10 times a day) also increase the occurrence, i.e. the probability goes down but the frequency goes up. Although this is not important for our results, it is issues that should be thought of.

The largest issue however is with regards to the number of unauthorized access attempts from outsider and the success rate of such attacks. As mentioned in Chapter 6 and Appendix G, these figures are most likely "wrong". The success of such attack depends on several elements, but they are all based on the how successful the social engineering attack is. Although the selected success rate and frequency of such attacks are considered uncertain, the usage of such figures becomes possible since our intentions have never been to perform an actual comparison. However, the fact the this event is so rare (rare event), throws light on a general issue with regards to

[8]The doctor must physically go to the ID office in order to reset the PIN code or get a new card

simulating security measures and security incidents/threats, namely that it becomes difficult to both include day-to-day activities, and rare events. Although this has little effect in our particular case, allowing such (rare) events to occur can become relevant for the final decision.

7.3 Discussion

Similar to the connection between IT decisions and KPIs which has been determined by Bartolini et. al ([5]) and Aiber et al. ([1]), our goal has been to identify a similar connection between security decisions and KPIs. That is, we are to determine the suitability of KPIs for measuring the effects a security decision has on a organization. By utilizing KPIs as "measure of performance" in the simulation, we have during the case study been able to identify how each of the security measures effects the KPIs. While no additional cost was identified with regards to the time used on the business activities, we did observe a difference between the two security measures with regards to number of deviations. In fact, we observed a clear difference with regards to which of the two security measures avoided *unauthorized access attacks*. Furthermore, the fact that we observe fewer authentication failures with smart cards should affect the work load for the employees. The work load could be related to the perception of convenience, and as stated by O'Gorman ([60]), a convenient authenticator will reduce the administrative costs. However, since monetary cost was not included in the scenario, we will not be able to confirm this statement. However, one needs to remember that when an authentication failure first occurs with smart cards, the cost for the employee with regards to time is higher. This might explain why the time related KPI indicates that the two security measures does not affect the business process different, i.e. over time, the two security measures should be equally time consuming for the employees. By having observed an effect on the KPIs by implementing two different security measures, we have simultaneously answered our first research question, namely that security decisions in fact does affect the goals of the organization.

All findings done during the case study are of interest since it provides us with information surrounding the effects a security measure has on the organization. However, further work must be done in order to verify these results. This is as mentioned earlier a step which is recognized as important in the methodologies described in [4], [46] and [35], but one which we, due to our objective, have not conducted.

Based on the conducted case study we have gained insight with regards to determining the suitability of the simulation approach as a method for analyzing the effects of a security decisions. Although we have made many simplifications during the construction of the scenario, the process of creating the model has still been a rather large and time consuming process. When further considering that we have reduced the magnitude of the process by not including validation and verification, it becomes evident that the time needed to build the model becomes very large. That said, by conducting a modeling and simulation approach, we are able to get insight into the system of interest by conducting "what if" analyzes. However, we need to compare the simulation approach with that of a non-simulation approach before we can make a final conclusion with regards to our third research question. Because of this, we will in Chapter 9 conduct a new case study based on a non-simulation approach in order to be able to compare the two approaches, and determine which is most suitable for analyzing the effects of security decisions.

8 A New Methodology for Model Design and Data Collection

In this Chapter we will present a new methodology which supports the process of creating a model used to simulate a security decisions effect on a set of identified KPIs. The methodology, called Minimalistic Model Design (MIMD), is based on the experience we have gained from the case study described in Chapter 7. The MIMD methodology is intended to replace the first step in the methodology described in Chapter 4, namely the *Design and data collection* step.

The overall motivations for creating the MIMD methodology is to reduce the complexity of the system processes, enable scalability through modularization, and identify and utilize indicators already used by the decision makers. The MIMD methodology consists of the following five steps, all with a focus on simplifying the system to be modeled to an acceptable level of detail.

1. Identify KPIs
2. Identify the object hierarchy
3. Identify the event hierarchy
4. Determine the mapping between identified KPIs, objects and events
5. Determine how objects responds to events

The main enabling factor which allows us to accomplish this is the exclusion of the temporal relationship within the business process, i.e. we do not model the actual process which is generated with the use of Business Process Modeling (BPM). As we will describe in more details later, the exclusion would be similar to excluding the *Sequence Diagram* found within the Unified Modeling Language (UML).

8.1 Motivations for creating a new methodology

During the case study, the realization of the demand for a methodology that reduces the complexity of the system became obvious. Although we will use the scenario as a example in this Chapter, the issues described in the example below is equally relevant in other types of organization.

In a hospital, several types of employees, including doctors, nurses and administrative personnel are employed. All employees will perform several different activities many times each day, and these sets of activities will all depend on the activities of the other employees. All activities are performed surrounding the treatment of the patients that are admitted at the hospital, and activities will to some extend affect the quality of the patient treatment. This small example quickly becomes complex in itself, but, in order to take into account both the security measures and the potential security threats, we need to expand the example further, and hence also the complexity increases. When employees perform most of their daily activities, they need to access sensitive information. In order to secure this information from unauthorized access, authentication mechanisms are introduced. An authentication mechanism can potentially affect the

performance of the activities negatively, e.g. if the security measures becomes too troublesome for the employees. However, a specific authentication mechanism can also have a positive effect on the performance of a activity compared to other types of authentication mechanism. Furthermore, one also needs to take into consideration how the security measures reduce the success rate of security threats, or mitigate the threats all together. When considering all these elements, how they interact with each other and the cause-and-affect of introducing a particular security measure to this already complex environment, the demand for a methodology that reduces the complexity of the real world becomes obvious.

Reducing the complexity of the real world system in order to be able to model it, is of course nothing new in the field of modeling and simulation. In order to reduce the complexity, all non-essential or non-relevant elements found in the business process, the security measure and the security threat should either be excluded or simplified, provided that this action does not affect the outcome. The reduction of the complexity becomes even more complicated in our situation. We need to combine several interacting elements[1] which are all complex in their own. Hence it becomes more difficult to determine when a model becomes too complex to handle and a model which becomes too simple, and hence does not provide us with our desired answers. The question which then naturally arises is that of "*how do we reduce the model sufficiently without affecting the outcome?*" This is a question which we believe we have answered with the MIMD methodology.

In addition, by enabling scalability, an organization can easily expand its modeled business process and the corresponding KPIs, and explore how several different security measures and threats, affect the organization. Such scalability is achieved by modularization, and the goal with enabling scalability is to be able to add new business processes, different types of security threats and security measures, either within the same category, e.g. smart card and password, or cross-category, e.g. authentication and backup, and determine how the different combinations affects the organization.

Finally, our beliefs concerning the benefits of utilizing indicators that are commonly used by decision makers when determining the effects of a security measure, have only increased. By utilizing KPIs when determining the effects of security measures, decision makers are presented with results that directly can be used in the decision making process. Because of this, the identification of KPIs has a dedicated step in the MIMD methodology.

All three demands, namely reducing complexity, enable scalability through modularization, and utilizing commonly used indicators, suggest that a high-level model is needed.

8.2 The MIMD methodology

The MIMD methodology, inspired by the modeling and simulation methodologies described in Chapter 2 and the experience gained during the case study (Chapter 7), consists of five steps. The process of developing the MIMD methodology has been an iterative process where the methodology has been altered several times during the process of conducting the case study. Because of this, we briefly describe in Appendix J the earlier versions of the methodology, and why these were not considered to achieve our three demands.

[1] The business process, the security measure and the security threats

The findings done at each of the five steps can be divided into three levels, as illustrated in Figure 17. Combined, the data collected in each of these levels provide us with the data needed in order to build the model. The five steps included in this phase includes identifying the organizations KPIs, the objects and events, the mapping between the objects and events, and finally, how each object responds to the identified events. What is important to notice, as this is one of the key reasons why we are able to reduce the complexity sufficiently, is that we do not included the temporal relationship between the identified business activities. That is, the MIMD methodology excludes a level of details that otherwise would have resulted in a considerably more complex model. We make the simplifying assumption that this does not invalidate the model.

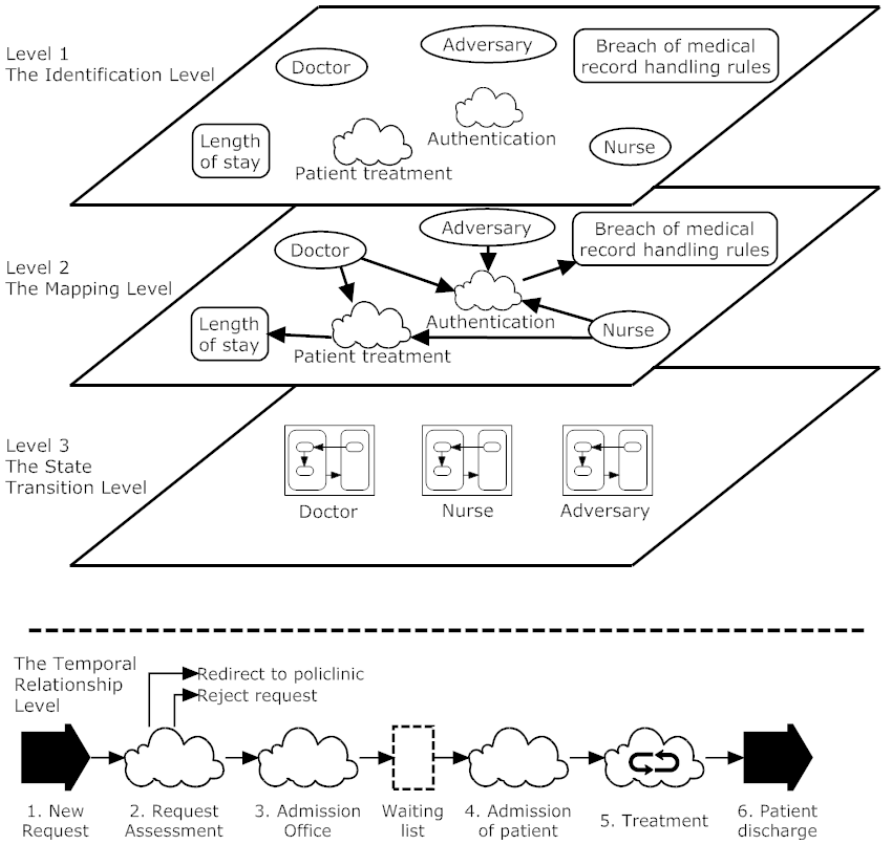

Figure 17: Illustration of the three levels of details utilized by MIMD

The first level, *The Identification Level*, identifies the "who" and "what", i.e. the relevant KPIs, objects and events. For example, this could be the "Length of stay" (KPI), a doctor (object) and authentication (event) in our hospital example. The second level, *The Mapping Level*, determines "who performs what". An example of such a mapping could for instance be the event of patient treatment which is performed by doctors and nurses. The "patient treatment" event would be

mapped to the "authentication" event through the common denominator of doctors and nurses. The third and final level, *The State Transition Level*, determines how each of the objects respond to a specific event and determines "What can happen". This level is needed in order to determine the behavior aspects of the model. One could argue that this level then incorporates the temporal relationship which we earlier stated had been excluded. However, this is not the case since we in this third level only determines how a single object response to events that is directly connect to the object itself. Should we have included the temporal relationship, as we define it, we would have been required to identify all interactions between events and objects, and how these again interacts with other such groupings. Such a temporal relationship would then have been added as a fourth level, but is as mentioned not included since we are able to draw sufficient amounts of data from level two and three, *The Mapping Level* and *The State Transition Level*, with regards to the interaction within the system in question.

To help the reader relate how we have simplified the process of modeling, let's consider how a similar simplification can be described with regards to the Unified Modeling Language (UML). Recall from Chapter 2 that UML was one of the visual modeling languages used to represent business processes. Our first level relates to UML's *Package Diagrams* where the systems objects are identified. Furthermore, UML's *Design Class Diagrams* (DCDs) are closely related to our second level. Similar to a DCD which include all relevant methods of a class, we identify relevant connections between the identified elements in a system. Another important similarity between DCDs and our second level is that neither of the two are concerned with the temporal relationship between the methods and events, respectively. That is, a DCD include all methods that a specific class can utilize, but the sequence of the execution of these methods is not of relevance. In both our methodology and in UML, the utilization of state transitions are similar. As mentioned earlier, a important reasons for enabling the reduction of the complexity of the model is the exclusion of the temporal relationship between activities in the business process. With regards to UML, this exclusion would be similar to excluding the *Sequence Diagram*.

Hopefully, it is now clear both what is included and what is excluded in the MIMD methodology. Furthermore, the motivation for excluding the temporal relationship should be clear, namely that it becomes to time consuming to include it. As mentioned before, we are only able to remove this relationship since we are still able to gain the needed insight with regards to how a specific security decision effects the organization.

The five steps included in the MIMD methodology will now be described in more details.

8.2.1 Step 1 - Identify KPIs

Identifying the KPIs relevant for the particular organization is an essential part of reaching our final goal. By identifying the KPIs we are also able to better determining which events are relevant, and which can be discarded early in the process.

In order to identify the KPIs, interviews with people within the organization becomes an important method of data collection. Furthermore, annual reports will contain much of the needed data. One can also identify the KPIs based on own experience and knowledge.

8.2.2 Step 2 - Identify the object hierarchy

As with the identification of KPIs, conducting interviews or otherwise contact people within the organization is a relatively cost-effective method of gaining insight into which objects that are relevant. Furthermore, one can also based on Business Process Diagrams (BPDs) of the business process in question get the needed insight. Risk analysis performed by the organization will also contain relevant information, e.g. with regards to who the adversaries are. Time sheets and job descriptions will provide information about the employees, while literature and vendor specifications provide us with information regarding equipment and software.

Based on the described sources, we can create the "*Object Tree*", which includes those objects that are relevant for the purpose of the model and simulation, e.g. employees, patients and systems.

As many objects most likely will share several properties, a object can inherit properties from their parent node. Such a solution is beneficial for several reasons, where scalability and structuring of the information are two of those reasons.

8.2.3 Step 3 - Identify the event hierarchy

The process of identifying relevant events is conducting in much the same way as with the former step, namely conducting interviews, analyzing the business processes and determining which events are identified in the risk analysis. Incident logs can also be utilized in order to identify former incidents.

Similar to the "*KPI Tree*" and the "*Object Tree*", we also create an "*Event Tree*" which list the identified events and potential sub-events. The events are divided into three levels, namely business activities and events, security measure events, and security threats and vulnerability events. Within these event categories, we find e.g. patient treatment processes, authentication failures and social engineering, respectively.

8.2.4 Step 4 - Determine the mapping between identified KPIs, objects and events

After having identified which objects and events are of interest, we need to answer the "who does what" question by determining the mapping between the objects and the events. The mapping also determines which events affects the identified KPIs. In addition to identifying "who does what", we also determine how long each specific task takes to conduct and how many times during a specific period, e.g. a work day, such a task is performed by the object. This could for instance be the task of approving documents, which is conducted by a doctor about 10 times per day, where each takes about 3 minutes to complete. Identifying the vulnerabilities of the system, the threats that can exploit these weaknesses, and the consequence of a successful attack, is also important to determine as this helps us to create a adversary model.

Again, the easiest method of answering such questions is by interviewing or otherwise contact people within the organization and gain access to the documentation from the risk analysis.

8.2.5 Step 5 - Determine how objects responds to events

The final step in the MIMD methodology is to determine how each object responds to a specific event. A state machine like interaction is created which determines the behavior of the model. An example would be to determine what a doctor will do if he is unable to authenticate himself

when needing to access a medical record, which would be to contact the ID office in order to reset the password[2].

8.3 Structuring and utilizing the collected data

The input parameters which are collected during the five steps in the MIMD methodology are used in the model. As most of the data collected will be based on the SME opinions of those interviewed, we have chosen to utilize the *triangular distribution* when applying the data in the model. *Triangular distributions*, as described in Appendix E, are often used in business decision making when data is scarce, or when the cost of collecting other sources of data becomes too high. The ability to utilize data which is not "complete" increases the usability of the methodology. The *Triangular Distribution*, as the name implies, utilizes three different values, namely the minimum, maximum and most likely (modus) value, for determining the distribution.

If available, the input parameters should be based on data from e.g. a risk analysis performed by the organization. In the health sector, the probabilities determined when the acceptance criteria's have been mapped can be utilized. Such figures help build confidence in the model and increase the chances of gaining the needed insight in order to make the "best" security decision.

In the MIMD methodology, the statistical distributions to be used in the model are separated into two categories. The first category covers distributions that are environment specific, i.e. data that in our particular case is health care related. This category is defined as *Environment Specific Distributions (ESDs)*. The second category covers distributions that are more general security related and defined as *Security Related Distributions (SRDs)*. Examples of distributions within this category is the number of authentication failures or number of attacks by a adversary. The separation between environment specific and security related distributions are done both because it helps to structure the data, but more importantly, because it helps us achieve the goal of scalability through modularization. As mentioned earlier, when comparing the different security measures we utilize Common Random Numbers (described in Appendix E). However, by utilizing the same ESDs, and as many similar SRDs as possible when comparing the two simulation runs, we further increase the precision of the comparison.

Both ESD and SRD are provided in the syntax of "*min, max, modus*" and the function for populating the model is defined as

$$\mathrm{SimRun}_a(\mathrm{SM}_z, \{[\mathrm{ESD}_{x1}], [\mathrm{ESD}_{xn}]\}, \{[\mathrm{SRD}_{y1}], [\mathrm{SRD}_{ym}]\}) \tag{8.1}$$

where SM is the type of security measure.

8.4 Developing a model based on modules

Based on the data collected during the five steps of the MIMD methodology, a model based on modularization can be created. Such modularization will both enable scalability with regards to ease of adding new modules, but also enables delegation of tasks between modelers within different fields of expertise. This is possible as a common, predefined, interface between the mod-

[2]This is at least normal procedure, and one can in some situation also experience that a doctor borrows another doctors password or similar to complete the task. Such an deviation would naturally be unacceptable, but it is still important to take such events into consideration.

ules is established. Based on such an approach, three separate modules can then been developed based on our scenario, as illustrated in Figure 18.

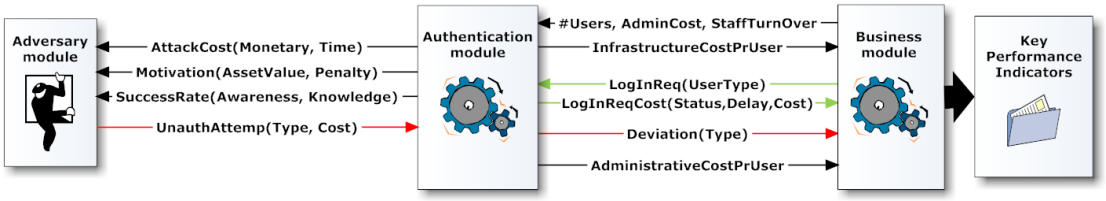

Figure 18: Illustration of the modularization

Within the *Business module* all business processes related issues are covered. This module also cover issues regarding the employees (users) and their interaction with the business process. Essentially, all non-security related issues are covered within the *business module*. Not surprisingly, the *Authentication module* includes those elements relevant for the two authentication mechanisms included in the decision making process. Finally, the *Adversary module* is utilized in order to determine what types of attacks occur, the frequency of such attacks, and what the consequence of the attack should it be successful. The interaction between the modules will be utilized in order to determine the behavior of the model.

Related to Figure 18 and the steps found in the MIMD methodology, each modules internal structure is determined by the mapping (step 4) between the objects and events, as well as the objects response to a specific event (step 5). The ESDs and some SRDs are found within the business module, while the rest of the SRDs are found in the authentication module. Within each module, both common and unique structure will be utilized. The hierarchic modeling structure is illustrated in Figure 19.

The common interface between the modules includes both monetary and non-monetary costs. The Business-to-Security measure-Interface (BSI) consists of ten different parameters, and the Security measure-to-Adversary model-Interface (SAI) consists of eight different parameters as listed in Table 16.

In the BSI, the authentication module is first initialized with the number of employees, the hourly cost of administrative tasks (*AdminCost*) and the turnover rate of the employees. Based on these three parameters, the *InfrastructureCostPrUser* is returned. Throughout the course of simulation run, a set of authentication requests will arrive from the business module, and based on the user type[3] the costs of the log in will be returned. If a authentication failure occurs, the cost of the log in request will return both monetary costs[4] as well as the actual delay[5]. Whether the authentication was successful or not is also included. All deviations which occur will be reported and the total administrative costs per user will be returned.

With regards to the SAI, we first determine the *resources needed* for an adversary, where money and time used are the parameters of interest. The second category, *motivation*, consists

[3]Different statistical distributions can be utilized based on the user.
[4]For instance based on the cost of resetting a password
[5]This delay involves the total time use on the authentication procedure

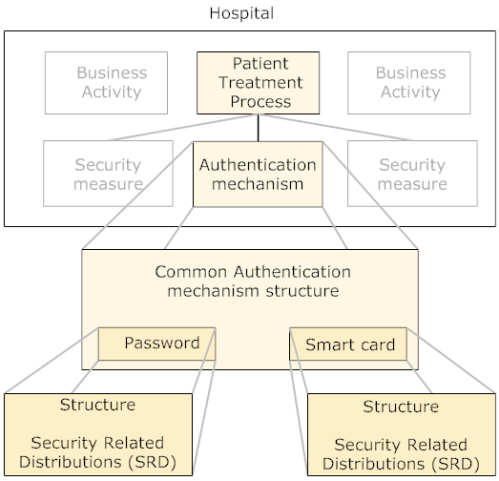

Figure 19: Illustration of hierarchic model structure

of parameters that include the value of the asset and the potential for criminal punishment. The third and final category, *success rate*, includes the awareness level of the users and the adversary's knowledge of the system. Combined, these six parameters helps us determining how likely an attack by a specific adversary is. That is, based on these six parameters, the type of attack and the costs of a successful attack can be determined.

Table 16: Parameters included in the module interface

	Parameters included in interface	
	BSI	**SAI**
1	#Users	AttackCostMonetary
2	AdminCost	AttackCostTime
3	StaffTurnOver	MotivationAssetValue
4	InfrastructureCostPrUser	MotivationPenalty
5	UserType	SuccessRateAwareness
6	LogInStatus	SuccessRateKnowledge
7	LogInDelay	UnauthAttemptType
8	LogInCost	UnauthAttemptCost
9	DeviationType	
10	AdministrativeCostPrUser	

It is important to notice that the all of eighteen parameters does not necessarily need to be included. Recall that the goal during step one of the MIMD methodology is to identify the KPIs which are of interest. Based on these KPIs, we can determine which elements need to be included in the interface, i.e. which parameters are populated.

8.5 Discussion

Banks et al. ([4]) have stated that model conceptualization is as much art as science. Although we agree in this statement, we have, based on experience gained during the case study described

in Chapter 7, developed the MIMD methodology which attempts to structure this process. In a information security context, where several independently complex systems needs to be combined, the MIMD methodology becomes particularly relevant as it helps to reduce the complexity of the system. It is important to notice that the simplifications performed in the scenario described in Chapter 6 are not the same as the complexity reduction described in the MIMD methodology.

The MIMD methodology consists of steps which helps us in the collection of relevant data and to help us identify the essential features of a problem for the construction of the model. In such the MIMD methodology is a replacement, or at least a supplement, for the *Design and data collection* step described in [4] and similar steps found in other methodologies. One of the largest discoveries, one which have made the MIMD methodology possible, is the exclusion of the temporal relationship between the different business activities. This exclusion allowed us to reduce the complexity of the system such that it simplifies the work process considerably. However, although we believe that such exclusion does not affect the final outcome, this needs to be confirmed in future work.

In addition to excluding the temporal relationship, the MIMD methodology also focuses on enabling scalability through modularization. We identified a similar approach by Aiber et al. ([1]), where the optimization process required a model of the system. The model was in [1] composed of three main sub-models, namely a business level model, IT model, and an IT to business level impact analysis model. Not unlike the model described in [1], we have also identified similar sub-models needed to fulfill our demand for modularization. In specific, with regards to the scenario we have determined a business module, a security measure module and a adversary module. In order to determine the interaction between the adversary module and the rest of the modules, the framework described by Olsen ([61]) has been utilized. Phase one in the adversary modeling framework described in [61] are covered in steps 2, 3 and 4 found in the MIMD methodology. The final two phases of the framework is covered by implementing the set of relevant security measures, and performing the simulation in order to determine how the organization is affected.

In order for the different modules to interact, a common interface has been identified. Again, a similar approach is found in [1], where the IT-to-business level impact analysis model enabled them to create a interface between two different levels of detail, which made it possible to align IT with the business processes. The interface parameters identified are both monetary and non-monetary, where the different parameters are selected based on the identified KPIs. The parameters infrastructure and administrative costs are based on the work of O'Gorman ([60]) and Helkala et al. ([27]). The relevance of these costs is confirmed based on the investment cost and operating costs described by Neubauer et al. ([49]). The resources needed and the motivation category is based on the work described by O'Gorman ([60]) and Wei et al. ([79]). The rest of the parameters are based on the adversary model framework described in [61] and the experiences gained during the case study described in Chapter 7. The parameters found in the interface between the security module and the adversary module are also in line with Kott et al. ([41]) definitions with regards to what determines an adversaries motivation for an attack. In [41], the probability of an attack is determined by the adversaries desired final result, his/her reasons for this result, and at what level of commitment the adversaries has to achieving the

goal.

Combined, this gives us confidence in that the selected parameters in fact are relevant and applicable. However, we have not been able to implement the modularization in our case study, and the interface hence needs to be validated, preferably through implementation. In addition, a sensitivity analysis needs to be conducted on these parameters in order to determine if more resources should be spent on some of the parameters in order to validate these. Furthermore, we have only identified an interface between the business module and the authentication module. Therefore, the interface between other modules should be determined to see if similarities exist.

9 Comparing the Simulation Approach with a Non-Simulation Approach

In this Chapter we will compare the approach of simulation with a non-simulation approach[1] in order to answer our research question of "To what extent is the construction and evaluation of a simulation model a suitable approach in determining how security decisions affect the goals of an organization?" The non-simulation approach selected for the comparison is the methodology described by Helkala et al. ([27]) which was identified in Chapter 2. We will refer to this methodology as the *Ranking methodology*.

In this Chapter we will first describe the Ranking methodology, and then apply this non-simulation approach on the scenario described in Chapter 6, before we finally compare the two approaches based on a set of criteria's.

9.1 Description of a non-simulation based methodology

The methodology described in [27], which we identified in Chapter 2, ranks authentications products and is further applied within the health sector. Based on this, the two approaches are applicable for comparison.

The ranking of authentication alternatives is carried out by defining a distance metric, i.e. product x is n units "better than" product y, and the methodology utilizes both a within comparison and a cross-category comparison method. This allows the methodology to both compare different types of authentication products and similar types of such products, e.g. different password policies.

The Ranking methodology evaluates a particular product based on four stages, namely *User and environment compatibility, Security level compatibility, Usability*, and *Costs*. When determining if a product fulfills the *user and environment compatibility* requirements, the product is compared to three different requirements, physical environment, device and user. The *Security level compatibility* is determined based on the entropy of the authenticator's search space ($H_{\mathtt{auth}}$) and the difficulty for an attacker to engineer a circumvention attack ($H_{\mathtt{circum}}$), where separate formulas for each category of authentication product is used. The *usability* of an authentication product is computed from the estimated annual time consumption per user for the different authentication activities. Finally, the *costs* are computed based on the infrastructure costs and the administrative costs.

A product which does not fulfill a specific stage will be excluded for further comparison. The sequence in which the stages are performed does not affect the output, but it is argued that this specific sequence is the most cost-effective as one can exclude products early on in the method by applying calculations and processing which requires less amount of work, compared to later steps such as the cost calculations. For more information about the Ranking methodology, see [27].

[1] That is, a analytical approach

9.2 A second case study - Scenario revisited with non-simulation approach

In order to compare the methodologies based on a set of criteria's (described later), we apply the Ranking methodology on the scenario[2] described in Chapter 6. The output from this second case study is found in Appendix L, while we will in this Section briefly describe the results from the case study.

9.2.1 Stage 1 - User and environment compatibility

In this first stage, both security measures were considered compatible with regards to the user and environment compatibility stage.

9.2.2 Stage 2 - Security level compatibility

The results from the security level compatibility stage yields that neither of the two products is secure, i.e. neither of them have sufficiently high entropy. We suspect that this result is due to the fictive figures used, and that these are not sufficiently correct. Similar security measures was in [27] identified as sufficiently secure. Because of this, we pay less attention to actual result, but focus rather on the fact that the two security measures have the same entropy[3]. According to the methodology, since none of the products are determined secure, we should stop at this point. However, based on the former argument, we continue with the rest of the steps in the methodology.

9.2.3 Stage 3 - Usability

The results from this third stage, as with the security level, yields that neither of the products are determined as usable products as both exceeds the threshold given by the methodology by a factor of two. We again see that the two produces have similarities, in this case as both have the same amount of delay. However, as with stage 2, although the methodology states that those products that are not determined as "usable" should not be further investigated, we continue with the next and final step since we suspect that the results are due to our fictive figures.

9.2.4 Stage 4 - Cost of infrastructure and administration

The results of the final stage, yields, perhaps not surprisingly, that the password scheme is less expensive than smart cards. In fact, according to this methodology, smart cards will be about twice as expensive, and about seven times as expensive if we don't including the "change passwords once-a-month-policy" figure, as using passwords. The final results can be seen in Table 17.

9.2.5 Discussion of the results

Although both of the security measures failed to fulfill the requirements of the Ranking methodology, we have seen that both security measures are quite similar, given the specific scenario. Both security measures received the same entropy (security level) and the same delay (usability). What became the deciding factor were the costs associated with each of the measures.

[2]Due to the difference between the two methodologies, we will need to utilize additional data which we have not used in our model. When possible, we will use collected but non-utilized data from our data collection phase, however where such data is not available, we will use data found in [27]. We do this as we believe that these figures are applicable in our case and that estimating these figures our self will not provide any better results.

[3]Given that one in the case of "Password" also uses a key in order to access the rooms in which the computers are available.

Table 17: Determining best authentication product with non-simulation approach

Authentication mechanism	Password	Smart Card
Infrastructure costs (€)	0	262,5
Administrative costs (€)	40,8 (108,8[1])	19,8
The total sum (€)	40,8 (108,8)	282,3
The total sum per user	3,4 (9)	23,5

[1]Although we have not included this in our model, one should in addition to the failed authentication attempts, also include the number of times a user is "forced" to chance the password. Assuming that the password policy requires that a user changes the password once a month and further assuming that such a password change takes only 1 minute, 12 minutes are added to see what effect this has on the outcome.

Based on the factor of costs, password was by the Ranking methodology determined as the "best" security measure[4].

Given that the results themselves are completely opposite of the results from the first case study, we need to discuss these findings more. The fact that cost was the deciding factor in the Ranking methodology, a factor which we in the first case study did not include, illustrates the impact the choice of ranking criteria's have. Monetary costs should without a doubt be included in the final decision making process, however, this should perhaps not be the final argument for the decision. That is, provided that an security measure is within a acceptable monetary cost, we believe that the ranking of the different security measures should be based on other measurements that determine the effect on the organizational goals. In such, if monetary costs had been included in our scenario, were both implementation costs and administrative costs had been included, but where both security measures had been within the acceptable monetary cost limits, the smart card would then still be the "best" choice. If the smart card product had been outside the acceptable monetary costs, then this security measure should have been eliminated from the final decision.

This suggests that the two methodologies can be merged in the sense that they both include important criteria's for the final decision. However, these issues do not affect our main goal as to determine the suitability of the two approaches.

9.3 Compare approaches based on predefined criteria's

As mentioned earlier, by comparing two methodologies that have taken different approaches, we are able to determine which of the two approaches are most suitable. It should be noted that, since the two methodologies are somewhat different with regards to the elements used in the analysis, the two methodologies cannot be directly be compared. However, we are able, based on the criteria's described below, able to draw some conclusions with regards to the suitability of each approach.

9.3.1 The criteria's

In this Section, the criteria's used in the comparison process is presented. The seven criteria's which are used enable us to determine the overall suitability of each approach. We chose to group several criteria's into a more general *work load* category since these reflect the combined

[4]Actually, the methodology concludes that none of the security measures are suitable given that they both fail the security level and the usability tests

work efforts needed for each approach. It is important to notice that the first criteria used are methodology specific.

1. Type of results provided
2. Possibilities for "what if" analysis
3. Degree of audit
4. Work load
   (a) Complexity of approach
   (b) Scalability
   (c) Type of input data needed
   (d) Computational resources needed

Each approach must produce results which is suitable for answering the question of which security measure is "best". Furthermore, it must be possible to determine if the results provided by the approach are reasonable, i.e. if we easily can validate and verify the results. Since it is desirable to determine how different system designs affect the organization, the approach's possibilities for enabling "what if" analysis is also a criterion which we are interested in determining. Finally, the result the approach provides should come at a relative low cost in order for the approach to be possible to be utilized in a real world situation. Because of this, the required *work load* will help us determine the approach's cost-effectiveness.

9.3.2 The comparison

We will in this Section, based on the identified criteria's in the former Section, compare the two approaches based on the conducted case studies. The comparison will be described below, but Table 18 summarizes the comparison.

Table 18: Summary of the comparison of the simulation and non-simulation approaches

		Result of comparison	
	Criteria's	**Simulation**	**Non-simulation**
1	Type of results provided	Affects on KPIs	Total monetary costs
2	Possibility for "What if" analysis	Good	Poor
3	Degree of audit	Weak	Good
4	Work load (hours)	73 + 250	73 + 32
5	Complexity of approach	High	Low
6	Scalability	Medium	Medium
7	Type of input data	KPIs, BAs, Objects and statistical distributions	Costs, Security level and Usability
8	Computational resources	Increasing	Constant

As we have already discussed, the first criteria is more of an methodology related issue, than the approaches themselves. As we can see from Table 18, the MIMD methodology focuses on how a security decision affects the identified KPIs. The Ranking methodology presents the results with regards to the total monetary cost of the authentication mechanism. Recall from earlier in this

thesis that KPIs can also include monetary costs, and in such, both methodologies can in fact determine the "best" authentication mechanism based on the same criteria. That said, the MIMD methodology allows more flexibility and is also indented to determine the effects of security decisions in general, and not only the authentication mechanisms effect on the organization.

A benefit with the simulation approach, is that it is possible to conduct in-depth "what if" analysis. Depending on the design experiments that are constructed, considerable amounts of data is possible to extract from the simulation run. This makes the simulation approach more flexible than the non-simulation approach.

In the case of the non-simulation approach, the results are them self more directly related from the input data, i.e. it is easier to determine how the results are processed. In the case of simulation, the processing of the input data is more difficult to verify manually. In such cases, one needs to be able to verify that the results them self are correct based on experience. This is a more comprehensive process as both the model and the results needs validation and verification.

The work load which is listed in Table 18 illustrates that the simulation approach is much more time consuming than the non-simulation approach. In fact, the figures suggest that the simulation approach is almost eight times more time consuming than the non-simulation approach. However, these figures needs some explanation. First of all, the time used on the non-simulation approach only includes the application of a already developed framework. In such, the total time used will increase if this was to be developed. However, the total time should still be lower than the simulation approach. Another remark is that, although the data collection phase conducted when creating the scenario where intended mainly for the creation of the simulation model, most of the data was also utilized in the non-simulation approach. Hence, we have in Table 18 separated the time used on the data collection phase (73 hours) and the actual simulation and modeling activities (about 250 hours). However, it is very important to notice that the 73 hours used on the data collection phase are only the time used on determining what type of data which is of relevance and gaining access to this data, and not the actual collection of this data. To illustrate, consider the input data of the authentication failures of smart cards. The data is based on one year of usage, however, we only include the time used to gain access to this data and process the data such that this can be utilized. Because of this difference, the actual data collection phase becomes much higher, although we do not include this in our comparison directly.

In both approaches, the complexity increases as the scenario increases in size. The simulation approach, with the application of the MIMD methodology, is able at handling the added complexity with regards to the scenario.

Scalability, with regards to comparing several different security measures, i.e. more than two, is better handled by the non-simulation approach since one can utilize more or less the same methods. In the case of simulation, each measure must be modeled and one needs to determine the simulation setup based on each new security measure. Hence, with regards to scalability, the non-simulation approach becomes better. This is at least the case in our implementation of the simulation approach and it is therefore also suggested in Chapter 12 that other implementation/methods are utilized.

The input data needed in both approaches are in fact quite similar. There are some differences, but these are mostly based on the methodology which utilizes the approach, not the approach

itself. The two methodologies both yield results that are applicable in determining which security measure is considered "best". The Ranking methodology presents the results with regards to monetary cost, but includes other important elements into consideration. The methodology present in this thesis focuses on using KPIs when determining which security measure is considered "best". By applying KPIs, which are determined by the organization, one has greater flexibility with regards to which elements that are of relevance. These can be cost-based, but does not need to be. The increased flexibility makes it possible to provide results in a format which directly answers the questions of interest. Basically, the results can be better aligned with what actually is of interest.

With regards to computational resources needed, the non-simulation approach remains constant while the simulation approach increases as the complexity increases.

9.3.3 Conclusion

When making our conclusion, we assume that in most situations one is required to make relatively fast decisions based on a narrow, or specific, area. That is, one needs answers to specific questions, and these questions needs to be answered as quickly as possible. Based on this assumption, our comparison indicate that the non-simulation approach is the most suitable approach, as this is the most cost-effective with regards to providing the needed answers with the least amount of work needed[5]. It is however important that the non-simulation based methodology then utilize a measure of performance which has the right focuses with regards to how the security measures affects the organization in question. The simulation approach is however, due to the added flexibility and possibility of extracting more information from the analysis, suitable if one needs to perform more thorough "what if" analysis.

In order words, the non-simulation approach provides us with the answers of a specific question, while the simulation approach provides us with a deeper insight and provides additional answers. In such, one needs to determine what type of approach best suits the desired outcome. This is also identified in the preparation phase of the methodology in Chapter 4 as a important decision to make early on in the process. One could then also conclude that one should first conduct the non-simulation approach, and if needed, further conduct the simulation approach.

Regardless of which approach one chooses, it becomes equally difficulty to gain access to the needed input data. Each approach will only process what they receive, and in such, if the input data are based on wrong assumptions, the results will be thereafter. Collecting "good" data becomes very difficult, which is especially true in the area of information security. However, with the non-simulation approach, we achieve a higher degree of audit in the results. That is, it is easier to make conclusions with regards to the validity of the results. Because of this, it is more likely that the results of the non-simulation approach is included in the decision making process, which after all must be considered the ultimate goal.

9.4 Discussion

The process of comparing the simulation approach with a non-simulation approach was performed in order to gain insight into the suitability of the simulation approach with regards to

[5] By comparing other methodologies utilizing the same approaches might have provided different results

determining the effects of security measures. However, although we have gained such insight, the suitability of the criteria's themselves is not determined. That is, the criteria's are subjectively selected, and although we have confidence in these criteria's, they are not compared to the work of others.

Another issue with the comparison process is that we have only compared the two approaches based on a single scenario. Since other type of scenarios could have yield quite different results with regards to the suitability, we strongly recommend that the approaches are compared based on other types of scenarios.

A final issue is that the results of the comparison, and hence the conclusions, are based on only two methodologies. We hence don't know if other non-simulation based methodologies would yield different conclusions. However, there are a couple of elements which increase our confidence in the particular methodology used in the comparison. First of all, the Ranking methodology is an extension of the work described by O'Gorman ([60]). The fact that the work is based on that of others should increase the quality of the methodology. Secondly, the comparison of the Ranking methodology made by Helkala et al. with regards to other related work identified in [27], further strengthen our confidence in the Ranking methodology. Based on these reasons, and the fact that the Ranking methodology is applied within the health sector and focuses on ranking authentication products, strengthen our beliefs that the methodology is a good foundation for the comparison of the simulation approach against non-simulation approach.

Although there are some issues surrounding the comparison process, we still believe that we have gained insight into the suitability of the simulation approach. As seen in Table 18 the non-simulation approach is the more cost-effective of the two. Furthermore, the non-simulation approach provides the decision makers with the highest degree of confidence in the results as there is a better connection between the input data and the results. Such a connection is important and is directly linked to the last challenge described in Chapter 2, namely that it is difficult to get acceptance of the results from the simulation process. Because this is crucial in order for the results to be utilized in the actual decision making process, the non-simulation approach becomes the preferred approach.

10 Summary of Contributions

In this Chapter we present the contributions made during the master thesis.

10.1 Scenario based data set

Based on the data collection phase conducted during the creation of our scenario, a uniformly scaled data set has been created. Since the data set for most parts are based on real systems, it allows others to utilize the data as input in similar models, or when conducting risk analysis. The data can either be utilized directly to save time during the data collection phase, or one can utilizing the data for comparison purposes.

10.2 Comparison of smart cards and passwords

A case study have been conducted to determine the effects different security measures have on a organization, and where the analysis of the results can be used in the decision making process. The case study applied a modeling and simulation methodology on a scenario which focuses on how the introduction of smart cards as a authentication mechanism, as opposed to passwords, affect the patient treatment process found within a Norwegian somatic hospital. The scenario was modeled and a simulation run was conducted for each of the security measures. The results indicated that the smart cards is favorable with regards to a lower amount of deviations, while no difference was detected between the two security measures with regards to any time related effect on the business activities. Based on the identified KPIs and the results provided by the simulation, smart cards are hence concluded to be the best security measure of the two.

10.3 A new methodology for model design and data collection

Several methodologies for modeling a system exists, e.g. those found in Banks et al. ([4]) and Law ([46]). However, based on experiences gained during the case study, we have developed supplement to these methodologies which helps to reduce the inherent complexity found within real life systems. The new methodology, called Minimalistic Model Design (MIMD), is intended to replace the *Design and data collection step* found in the methodology described Chapter 4. In such, the MIMD methodology is not a complete modeling and simulation methodology, but rather an addition to existing methodologies.

The MIMD methodology have been designed with a focus of achieving three main goals, namely to reduce the complexity of the interacting elements of the security measures and the business process, to enable scalability through modularization, and finally, to utilizing indicators which are commonly used by decision makers. The methodology achieves these three goals by identifying relevant KPIs, objects and events within the system. The mapping between the objects and the events are then established, and the methodology then determines how the objects respond to different events. A crucial element in achieving the goal of reducing the complexity is to not include the temporal relationship between the identified business activities. That

is, the MIMD methodology excludes a level of details that otherwise would have resulted in a considerably more complex model.

Based on the modularization process, a common interface has been identified. The Business-to-Security measure-Interface (BSI) consists of ten parameters, namely *#Users*, *AdminCost*, *Staff-TurnOver*, *Infrastructure cost*, *UserType LoginStatus*, *LogInDelay*, *LoginCost*, *DeviationType* and *Administrative costs*. The Security measure-to-Adversary model-Interface (SAI) consists of eight parameters, where these are divided into three categories, *AttackCost, Motivation* and *Success Rate*. Based on these three categories, different types of attacks are returned and the cost should such an attack succeed.

10.4 Utilization of the Ranking methodology

In order to compare the simulation approach with that of a non-simulation approach, we applied a non-simulation approach described by Helkala et al. ([27]) to the scenario described in this thesis. The Ranking methodology determines the best authentication products by following a four step process. Based on the four steps, the methodology determines a products user and environment compatibility, security level, usability, and costs.

There were some issues surrounding the outputs from the methodology due to our input data, but we were able to observe that both security measures received very similar security and usability levels, but that passwords was ranked as the best security measure based on the cost-based focus of the methodology.

Although this case study was conducted as a secondary goal, where the primary goal was to be able to compare a simulation based approach with that of a non-simulation based approach, we have still conducted a case study with a methodology which few others have utilized before us. Because of this, the results from this case study are hence also recognized as part of our contribution.

10.5 Comparing the simulation approach with a non-simulation approach

In order to determine if the simulation approach is suitable for determining how a security decision affects an organization, we need to conduct a comparison. We have therefore conducted a comparison between the simulation approach described in this thesis with the non-simulation approach developed by Helkala et al. ([27]). Based on a set of criteria's which we have defined and the experience we gained from the two case studies, we were able to compared the two approaches.

The criteria's defined focused on determining which possibilities the approach had for "what if" analysis and further determined the work load associated with each of the approaches. Furthermore, although more methodology specific, the criteria's also determined which type of results was provided.

Based on the comparison, a two-folded conclusion was reached. As the non-simulation approach is more cost-effective than a simulation based approach, and in addition provides results which are more related to the input data, this approach is recommended. That said, we also recognize that the simulation approach is recommended if one needs to conduct a "what if" analysis due to the added flexibility and possibility of extracting more information from the analysis.

11 Discussion

In this Chapter we will extend our former discussions based on the experiences made during the writing of this thesis. We will describe how the work done in this thesis related to our three research questions, the related work described in Chapter 2 and the challenges identified within these.

As mentioned in Chapter 1, the purpose of information security, and information technology (IT) in general, is to support the business processes in such a way that the organization achieves its goals. Hence, information security, and security controls in specific, must be considered in the context of the business in which they operate. In fact, the demand for a tool which allows security managers to make consistent security decisions was identified by Butler ([8]). Therefore, one of our goal with this master thesis have been to determine if simulation is a suitable approach for analyzing the effects security decisions have on a organization. Furthermore, we also need to determine if information security decisions in fact do influence the organizational goals. In order to measure such effects, we need to utilize a measures which incorporates those aspects of organizational performance that are the most critical for the current and future success of the organization. Therefore, we have utilized KPIs and determined if they were suitable for determining the effects of security decisions.

In order to answer our research questions, we first selected the simulation method of Discrete Event Simulation, as this was identified as the best choice with regards to simulating business processes. In fact, it is determined by Laguna et al. ([43]) that simulation in the context of process modeling actually refers to discrete computer events simulation. We then selected the modeling and simulation methodology described by Banks et al. ([4]) as our basis for the thesis. The choice was made by comparing this methodology with those methodologies and processes described in [46], [35] and [11]. Based on this comparison, we have confidence in that the methodology applied in this thesis is a solid foundation for answering our research questions. In addition to selecting a methodology, we have also conducted a selection process based on the criteria's described in [4] in order to determine which simulation environment suites our task the best. Although there were some issues surrounding this selection process, namely that few environments was possible to test and the choice of environment was based on a subjective evaluation, we still believe that a suitable environment have been selected. In retrospect, based on the specific scenario which was selected, it might have been easier to select a simulation environment with a specific health care package. However, since the methodology is not simulation environment specific, and the choice of environment will not determine the suitability of the simulation approach in general, the choice of simulation environment is less important.

In order to answer our research questions, a health care specific scenario was created in collaboration with Ahus, the Rheumatism Hospital and Buypass. In the scenario, the security decision of which authentication mechanism to implement was in focus. Access to data was one of the largest reasons for developing a scenario in collaboration with the aforementioned

organizations. Because of this, we have been able to construct a scenario which mostly is based on data from a real system. Unfortunately, not all of the required input data was possible to obtain from the organizations. This hence also resulted in some issues surrounding the validity of this specific input data. In order to compensate for the uncertainty of the validity, other sources was utilized in an attempt to gain a higher confidence in the utilized data. That said, some of the data still includes a large degree of uncertainty, and in such some of the results must be treated thereafter. In addition, several simplifications were needed to be made with regards to the scenario in order for us to later model this system. Although we believe that those simplifications made will not affect our final goal of determining the suitability of the simulation approach, the results of the simulation runs must be treated based on the fact that we have not validated all of the data, nor verified the model which has been developed based on the simplified scenario. The issues surrounding the lack of validity of the data and the model structure is however acceptable since our goal is to determine the suitability of combining the simulation approach with the usage of KPIs, and not perform an assessment for Ahus. The analysis of the simulation run data was conducted based on a methodology which utilizes steps recognized by several sources as important.

Regardless of the aforementioned issues, the case study which was conducted based on the selected methodology and the created scenario, enabled us to answer two of our research questions in addition to some of the challenges identified. The results of the case study indicated that the two security measures affects the business processes equally with regards to time used, but that smart cards is the best choice with regards to the number of deviations observed. That is, the two measures become equally user friendly, which also was determined with the Ranking methodology, but with regards to security, the results indicate that smart cards are "best".

Based on these results we have also been able to determine the suitability of KPIs, since we in fact are able to analyze the effects of the security decision. By using KPIs as the measure of performance in the simulation, we have also provided insight into the challenge describe by Bartolini et al. ([6]), namely that one needs to link performance optimization metrics with KPI. Although we have not attempted to answer the challenge described by Neubauer et al. ([49]) directly, i.e. determine which approach is most suitable for determining the consequences of security issues with regards to loss of reputation, we have gained insight into this challenge during this thesis. That is, the deviation related KPIs are linked to the acceptance criteria's and the incidents which could lead to loss of reputation [54]. Such a connection allows us to identify which security measures results in deviations which again could lead to loss of reputation. However, as stated, we have not compared simulation with other approaches with regards to answering this challenge, and in such, more work needs to be done. The results of the case study illustrate that we are able to determine that the implementation of smart cards reduces the expected loss compared to implementing passwords. Such insight was identified by Gordon et al. ([23]) as the key in analyzing information security decisions.

In addition to determining the suitability of the KPIs based on the results of the case study, we have simultaneously answered our research question of whether information security decisions in fact influence organizational goals. However, since only one scenario has been utilized, we cannot make a final conclusion with regards to the suitability of KPIs nor information security

decisions' influence on the organization. For example, since the scenario did not include monetary costs, we have not been able to include such KPIs in the analyses of the suitability of the KPIs. Furthermore, we are not able to determine how the information security decisions in scenarios outside the health sector, or in other health care specific scenarios, affect the organization. That said, based on the results, we do make the preliminary conclusion that information security decisions in general influence organizations goals and that KPIs are suitable for detecting these effects. This conclusion is also based on the related work which we have identified, e.g. Aiber et al. ([1]) who identified a clear connection between IT related policy decisions and business level metrics such as profit or ROI.

Although the case study gave us much insight into the simulation approach, we could however not answer our final research question purely based on this case study. In order to be able to make such decisions, we needed to compare the simulation approach with that of a non-simulation approach. Because of this, we conducted a new case study based on the same health care specific scenario but with the use of a non-simulation approach. By comparing the processes surrounding the two case studies, we were in fact able to answer our main question of whether the simulation approach is suitable. Recall that a great deal of work was required with the simulation approach, and this work load did not include a full scale scenario or the validation and verification phases. The total amount of work needed when conducting the simulation approach, compared to the non-simulation approach, becomes one of the largest drawbacks of the simulation approach. The non-simulation approach becomes more cost-effective, and provides a more direct relationship between the input data and the results. The challenge described by Holm et al. ([29]) concerning the difficulty of getting decision makers to trust the results of the simulation approach makes the direct relationship important. Furthermore, considering that fast decisions are needed surrounding some specific criteria's, the non-simulation approach is considered best. However, the simulation approach becomes suitable, more so than the non-simulation, when in-depth "what if" analysis is needed to be conducted. Based on this, we therefore conclude that a non-simulation approach should first be conducted, and if necessary, one can conduct an in-depth "what if" analysis based on the results of the non-simulation approach. It is however important to notice that, regardless of which of the two approaches are selected, the process of data collection is equally difficult.

Should an in-depth "what if" analysis be necessary to conduct, we have, based on our experience during the first case study, developed a methodology which helps the modeler in the collection of input data and in reducing the complexity of the system of interest. The methodology, called Minimalistic Model Design (MIMD), reduces the complexity by removing the temporal relationship between the identified business activities. Furthermore, the methodology results in a modularization based model, were a common interface between the modules have been identified. The parameters found within the interface are used to determine how the authentication mechanisms affect the business module. In such, the interface provides insight into the challenge identified by Cohen ([12]), namely that there are no set of commonly accepted metrics upon which to base a set of measurements to be used for simulation. However, the interface, although based on several other sources and our own experience during the case study, needs to be implemented in order to be validated. Furthermore, since we have only determined the interface based

on the authentication module, additional work must be done to determine if similar interfaces can be determined for other security related modules.

Although we have been able to answer our research questions during this master thesis, there have been some issues during the process of answering these questions. Especially, the construction of the model has been a time consuming process where several large alterations have been necessary. Based on feedback and our own realizations, we have gone from a detailed model which incorporated all elements of a business process, to a model which only focuses on the high-level objectives and events of the system of interest. Although some of these issues are related to the model experience of the authors, it also confirmed some of our suspicions with regards to the suitability of the approach. Since much uncertainty exists in the input data, the results must be treated thereafter. Because of this, the cost of the process involved in the approach, from beginning to end, needs to be justifiable. That is, since the results most likely will only be used as a supplement in the decision making, these results must be based on a cost-effective process - which we have concluded that the simulation approach is not.

12 Future Work

In this Chapter we will present the recommended future work based on our findings made during this master thesis.

In the MIMD methodology, the exclusion of the temporal relationship between the identified business activities has been the enabling factor with regards to the complexity reduction. Although we believe that such exclusion does not affect the final results, further work must be done in order to confirm or disprove these assumptions.

Furthermore, the modularization approach described in the MIMD methodology, and the interface which exists between the modules, needs to be implemented. That is, the methodology we have developed for others to use must be tested. In addition to validating (or disproving) these parameters, other types of modules should be included, i.e. other security measures needs to be included. By including other modules, one is able to determine if the identified interface is common for other types of security measures as well. If this is the case, the process of simulating the effects of security decision would become considerably simplified.

Furthermore, a sensitivity analysis should be conducted in order to determine which of the input parameters we need to focus our efforts on. That is, should the value of a small number of parameters greatly influence the results, we must spend most resources on the collection of valid data with regards to these sensitive parameters.

Since we have simplified the scenario by only incorporating doctors and two security measures, the scenario, and hence the case study, could be expanded. One could incorporate how the results are affected by taken into consideration the distributed locations at Ahus, were persons who need to visit the ID office in fact needs to travel considerable distances. Furthermore, although we have determined how the KPIs are affected by the security decisions, we have not included how the users are affected by such decisions. In such, the scenario should be expanded to incorporate more of the complexity which in fact exists in the system, such that the MIMD methodology truly can be tested. One approach for gaining the needed data would be to perform a survey within Ahus to better understand the affects the security measure implemented have had on the employees. Such a survey could increase the sophistication of the input data we have used in our case study. Since we have not confirmed the validity of our case study results, such an expansion of the scenario would help to determine if our assumptions have been correct, and hence, if our results and conclusions are valid.

As mentioned in Chapter 9, an issue regarding the validity of the process of comparing the simulation and non-simulation approach is related to the fact that we have only compared the two approaches based on a single scenario. We therefore strongly recommend that the approaches are utilized based on other types of scenarios in order to determine if this affects our conclusions. For example, one could determine how different user/security training programs or security policies actually affect the organization, or how the introduction of a VoIP system with

and without security measures could affect the organization. As mentioned earlier, the incorporation of the human factor is also important and should be tested.

Another suggestion of future work is to apply the simulation approach in combination with the MIMD methodology in a full scale assessment of the effects of security decisions on a organization. That is, apply the approach and the MIMD methodology on a organization that is to make a set of security decisions. By utilizing the results from the simulation in the decision making process, and then later review the actual impact the decision have had on the organization, one is able to determine the validity of the results. Such a thorough and time consuming approach would fall most natural for a PhD project, although it can also be done over two master theses, each of them covering a specific area, i.e. one applying the methodology and the other reviewing the actual impact of the security measure.

In Chapter 2 a demand of an automated tool for making consistent security decisions was identified. However, the approach taken in this thesis does not fully comply with this demand since it is not possible to compare multiple ($k > 2$) decisions at once in a cost-effective way. We therefore suggest that the recommendations that Swisher et al. ([73]) makes with regards to combining the Ranking & Selection approach and the Multiple Comparison with the Best approach should be made. Recall from Chapter 2 that combining these approaches would both enable us to select the best system and draw inferences about the relationships between the systems.

13 Conclusions

We will in this Chapter make our final conclusions with regards to our research questions.

The results of the first case study conducted in this thesis indicate that security decisions do influence the organizational goals, and the KPIs are suitable of detecting the effects of the security decisions. In particular, the case study provided us with results which determined that there where no statistical significant differences between the two security measures with regards to the time based measure of performance. However, we did determine that smart cards are the preferred security measure based on the reduction of deviations. In fact, with smart cards, unauthorized access from outsiders was mitigated. To summarize the results of the case study we have seen that the increased security level which smart cards provide does not negatively affect the business activities conducted by doctors.

When comparing the simulation approach with the non-simulation approach, the latter approach was determined the more suitable of the two. One of the reasons for this is that the non-simulation approach was more cost-effective with regards to providing the needed answers within a short time frame. This conclusion is also based on the non-simulations more direct relationship between the input data and the results, with becomes important in order for the results to be trusted by the decision makers. That said, the simulation approach is a suitable approach if a in-depth "what if" analysis is needed. This suggest that one should conduct the non-simulation based approach first, and if additional insight is needed, further analysis can be made with the simulation based approach.

Should the simulation approach be applied, the utilization of the MIMD methodology presented in this thesis will be able to reduced the complexity of the model. As described in the thesis, the reduction is conducted by excluding the temporal relationship between the identified business activities and is possible to conduct without significantly affecting the final outcome.

However, regardless of which approach is utilized, the process of data collection is equally difficult. This difficulty becomes perhaps even more evident within the field of information security since little information is available, and much uncertainty exists with the information which is available. However, as with risk analysis, the true benefit of the modeling and simulation activity might lie in the process itself and the realizations made during this process. That is, by "forcing" ourselves to determine a set of figures and identifying the important business processes, the increased insight and awareness of potential problems outweigh the uncertainties surrounding the validation of the input data and the model.

Bibliography

[1] S. Aiber, D. Gilat, A. Landau, N. Razinkov, A. Sela, and S. Wasserkrug. Autonomic self-optimization according to business objectives. *Proc. International Conference on Autonomic Computing*, pages 206–213, 2004.

[2] South-Eastern Norway Regional Health Authority. Årlig melding 2008 for helse sør-Øst rhf til helse- og omsorgsdepartementet, 2008. `http://www.helse-sorost.no/modules/module_123/proxy.asp?D=1&C=135&I=0&mids=a26a98a` - Last visited 28th June 2010.

[3] W. Bandara, G. G. Gable, and M. Rosemann. Factors and measures of business process modelling: model building through a multiple case study. *European Journal of Information Systems*, 14(4):347–360, 2005.

[4] J. Banks, J. Carson, L. B. Nelson, and M. D. Nicol. *Discrete-Event System Simulation*. Prentice Hall, 2004.

[5] C. Bartolini, M. Salle, and D. Trastour. It service management driven by business objectives - an application to incident management. *In: Proc. IEEE/IFIP Network Operations and Management Symposium*, April 2006.

[6] C. Bartolini, C. Stefanelli, and M. Tortonesi. Symian: A simulation tool for the optimization of the it incident management. *Managing Large-Scale Service Deployment*, 5273:83–94, 2008.

[7] A. C. Boynton and R. W. Zmud. An assessment of critical success factors. *Sloan Management Review*, 4(25):17–27, 1986. `http://as.nida.ac.th/~waraporn/resource/704-1-50/Readings/6-Assessment%20CSF-Boynton-Zmud.pdf` - Lasted visited 28th June 2010.

[8] S. A. Butler. Security attribute evaluation method: a cost-benefit approach. *ICSE '02: Proceedings of the 24th International Conference on Software Engineering*, pages 232–240, 2002.

[9] Y. Carson and A. Maria. Simulation optimization: methods and applications. *WSC '97: Proceedings of the 29th conference on Winter simulation*, pages 118–126, 1997.

[10] F. E. Cellier. Qualitative modeling and simulation: promise or illusion. *Proc. Winter Simulation Conference*, pages 1086–1090, 1991.

[11] M. A. Centeno and M. F. Reyes. So you have your model: What to do next - a tutorial on simulation output analysis. *Proceedings of the 1998 Winter Simulation Conference*, 1998.

[12] F. Cohen. Simulating cyber attacks, defences, and consequences. *Computers & Security*, 18(6):476–518, 1999.

[13] Dassault Systemes Corporation. Quest simulation environment. www.3ds.com/products/delmia - Last visited 28th June 2010.

[14] ProModel Corporation. Promodel simulation environment. www.promodel.com - Last visited 28th June 2010.

[15] Simul8 Corporation. Simul8 simulation environment. www.simul8.com - Last visited 28th June 2010.

[16] Visual8 Corporation. Automod simulation environment. www.visual8.com/automod.html - Last visited 28th June 2010.

[17] H. Damerdji and M. K. Nakayama. Two-stage multiple-comparison procedures for steady-state simulations. *ACM Transactions on Modeling and Computer Simulation*, 9(1):1–30, 1999.

[18] G. P. Dwyer Jr. and K. B. Williams. Portable random number generators. 1999. http://www.jerrydwyer.com/pdf/random.pdf - Last visited 28th June 2010.

[19] G. W. Evans, B. Stuckman, and M. Mollaghasemi. Multicriteria optimization of simulation models. *WSC '91: Proceedings of the 23rd conference on Winter simulation*, pages 894–900, 1991.

[20] Control Objectives for Business Information-related Technology (COBIT). http://www.isaca.org/cobit.htm - Last visited 28th June 2010.

[21] Kompetansesenter for IT i helse-og sosialsektoren (KITH). Indikatorer for informasjonssikkerhet, 2004. http://www.kith.no/templates/kith_WebPage____727.aspx - Last visited 28th June 2010.

[22] C. Forgia and R. Revetria. System dynamics and regressive meta-modeling applied methodology for improving management performances in services industry: a case study in supply chain and highway maintenance. *SpringSim '08: Proceedings of the 2008 Spring simulation multiconference*, pages 279–287, 2008.

[23] L. A. Gordon and M. P. Loeb. The economics of information security investment. *ACM Transactions on Information and System Security*, 5(4):438–457, 2002.

[24] H. Hasle, Y. Kristiansen, K. Kintel, and E. Snekkenes. Measuring resistance to social engineering. *Lecture Notes in Computer Science*, 3439:132–143, 2005.

[25] P. Heidelberger. Fast simulation of rare events in queueing and reliability models. *ACM Trans. Model. Comput. Simul.*, 5(1):43–85, 1995.

[26] K. Helkala. Authentication in norwegian health services (survey report). *Proceedings of International Symposium on Health Informatics and Bioinformatics*, 2007.

[27] K. Helkala and E. Snekkenes. Formalizing the ranking of authentication products. *Information Management and Computer Security*, 17(1):30–43, 2009.

[28] SINTEF Helse. Samdata somatikk sektorrapport 2005. http://www.helsedirektoratet.no/vp/multimedia/archive/00011/Rapport_SAMDATA_Soma_11544a.pdf - Last visited 28th June 2010.

[29] L. B. Holm and F. A. Dahl. Simulating the effect of physician triage in the emergency department of akershus university hospital. *Proceedings of the 2009 Winter Simulation Conference*, pages 1896 – 1905, 2009.

[30] Akershus University Hospital. Årsmelding 2008, 2008. http://ahus.no/modules/module_123/proxy.asp?D=2&C=1050&I=18471&mids=a2650a2675a - Last visited 28th June 2010.

[31] Akershus University Hospital. Årlig melding 2009 til helse sør-Øst rhf, 2009. http://www.ahus.no/stream_file.asp?iEntityId=20093 - Last visited 28th June 2010.

[32] P. Huang and K. Sycara. Learning from and about the opponent. In Alexander Kott and William M. McEneaney, editors, *Adversarial Reasoning: Computational Approaches to Reading the Opponent's Mind*, pages 315–333. Chapman & Hall/CRC, 2006.

[33] Flexsim Software Products Inc. Flexsim simulation environment. www.flexsim.com - Last visited 28th June 2010.

[34] Imagine That Inc. Extendsim simulation environment. www.extendsim.com - Last visited 28th June 2010.

[35] Imagine That Inc. User guide, 2007. http://www.extendsim.com/support_manuals.html - Last visited 28th June 2010.

[36] Rockwell Automation Inc. Arena simulation environment. www.arenasimulation.com - Last visited 28th June 2010.

[37] Balanced Scorecard Institute. Balanced scorecard. http://www.balancedscorecard.org/bscresources/aboutthebalancedscorecard/tabid/55/default.aspx - Last visited 28th June 2010.

[38] Information Technology Infrastructure Library (ITIL). http://www.ogc.gov.uk/guidance_itil.asp - Last visited 28th June 2010.

[39] S. H. Jacobson, S. N. Hall, and J. R. Swisher. Discrete event simulation of health care systems. *Patient Flow: Reducing Delay in Healthcare Delivery*, 91:211–252, 2006.

[40] A. Jaquith. *Security Metrics - replacing fear, uncertainty, and doubt*. Pearson Education, 2007.

[41] A. Kott and W. M. McEneaney. *Adversarial Reasoning: Computational Approaches to Reading the Opponent's Mind*. Chapman & Hall/CRC, 2006.

[42] L. Kvæl. Beregning av kostnadsvekter til den norske versjonen av drg-systemet 2005, 2006. http://www.co2sim.net/Home/Publications/Publication?page=25824 - Last visited 28th June 2010.

[43] M. Laguna and J. Marklund. *Business Process Modeling, Simulation, and Design*. Pearson/Prentice Hall, 2005.

[44] A. Landau, S. Wasserkrug, D. Gilat, N. Razinkov, A. Sela, and S. Aiber. A methodological framework for business-oriented modeling of it infrastructure. *WSC '04: Proceedings of the 36th conference on Winter simulation*, pages 474–482, 2004.

[45] Lanner. Witness simulation environment. `www.lanner.com/en/witness.cfm` - Last visited 28th June 2010.

[46] A. M. Law. *Simulation Modeling and Analysis*. McGraw-Hill, fourth edition, 2006.

[47] P. A. W. Lewis, A. S. Goodman, and J. M. Miller. A pseudo-random number generator for the system/360. *IBM Syst. J.*, 8(2):136–146, 1969.

[48] K. D. Mitnick and W. L. Simon. *The art of deception: Controlling the human element of security*. John Wiley & Sons, 2002.

[49] T. Neubauer, M. Klemen, and S. Biffl. Business process-based valuation of it-security. *EDSER '05: Proceedings of the seventh international workshop on Economics-driven software engineering research*, pages 1–5, 2005.

[50] Ministry of Health and Care Services. Key figures. `http://www.regjeringen.no/nb/dep/hod/tema/sykehus/nokkeltall-og-fakta---ny/nokkeltall.html?id=528647` - Last visited 28th June 2010.

[51] Ministry of Health and Care Services. Behovsbasert finansiering av spesialisthelsetjenesten, norges offentlige utredninger 2003:1, 2003. `http://www.regjeringen.no/nb/dep/hod/dok/nouer/2003/nou-2003-1.html?id=453861` - Last visited 28th June 2010.

[52] Ministry of Health and Care Services. Oppdragsdokument helse sør-øst, 2010. `http://www.regjeringen.no/nb/dep/hod/tema/sykehus/styringsdokumenter/oppdragsdokument.html?id=535564` - Last visited 28th June 2010.

[53] Norwegian Directorate of Health. Faktaark 12 - tilbakerapportering av resultater fra it-driften, 2006. `http://www.helsedirektoratet.no/samspill/informasjonssikkerhet/norm_for_informasjonssikkerhet_i_helsesektoren_232354` - Last visited 28th June 2010.

[54] Norwegian Directorate of Health. Norm for informasjonssikkerhet i helsesektoren, 2006. `http://www.helsedirektoratet.no/samspill/informasjonssikkerhet/norm_for_informasjonssikkerhet_i_helsesektoren_232354` - Last visited 28th June 2010.

[55] Norwegian Directorate of Health. Innsatsstyrt finansiering 2009, 2008. `http://www.helsedirektoratet.no/vp/multimedia/archive/00092/Innsatsstyrt_finansi_92249a.pdf` - Last visited 28th June 2010.

[56] Norwegian Directorate of Health. Faktaark 4 - kartlegging av klassifisering av systemer i henhold til kritikalitet i forhold til behov for tilgjengelighet, 2009. http://www.helsedirektoratet.no/samspill/informasjonssikkerhet/norm_for_informasjonssikkerhet_i_helsesektoren_232354 - Last visited 28th June 2010.

[57] Norwegian Directorate of Health. Faktaark 5 - fastsette akseptkriterier for tilgjengelighet, konfidensialitet, integritet og kvalitet, 2009. http://www.helsedirektoratet.no/samspill/informasjonssikkerhet/norm_for_informasjonssikkerhet_i_helsesektoren_232354 - Last visited 28th June 2010.

[58] Norwegian Directorate of Health. Faktaark 8 - avviksbehandling, 2009. http://www.helsedirektoratet.no/samspill/informasjonssikkerhet/norm_for_informasjonssikkerhet_i_helsesektoren_232354 - Last visited 28th June 2010.

[59] The National Institute of Standards and Technology. Security: electronic authentication guideline. *NIST Special Publication 800-63*, 2006.

[60] L. O'Gorman. Comparing passwords, tokens, and biometrics for user authentication. *Proceedings of the IEEE*, 91(12), 2003.

[61] O. K. Olsen. Adversary modelling. *Master's Thesis, Gjovik University College*, 2005.

[62] G. L. Orgill, G. W. Romney, M. G. Bailey, and P. M. Orgill. The urgency for effective user privacy-education to counter social engineering attacks on secure computer systems. In *CITC5 '04: Proceedings of the 5th conference on Information technology education*, pages 177–181. ACM, 2004.

[63] D. Parmenter. *Key Performance Indicators - Developing, Implementing, and Using Winning KPIs*. John Wiley & Sons, 2007.

[64] T. Popkov, Y. Karpov, and M. Garifullin. Using simulation modeling for it cost analysis. *The 10th HP Open View University Association Workshop, Switzerland*, 2003.

[65] Norwegian Post and Telecommunications Authority. Det norske ekommarkedet 2008. http://www.npt.no/ikbViewer/Content/111315/Ekommarked_2008_v13_rev5.pdf - Last visited 28th June 2010.

[66] R. Richardson. Csi computer crime and security survey 2008. http://gocsi.com/survey - Last visited 28th June 2010.

[67] SAP. Bpm technology taxonomy - a guided tour to the application of bpm, 2009. http://www.sap.com/community/showdetail.epx?ItemID=17407 - Last visited 28th June 2010.

[68] R. G. Sargent. Verification and validation of simulation models. *WSC '07: Proceedings of the 39th conference on Winter simulation*, pages 124–137, 2007.

[69] L. Schrage. A more portable fortran random number generator. *ACM Transactions on Mathematical Software*, 5(2):132–138, 1979.

[70] Alion Science and Technology. Micro saint simulation environment. www.alionscience.com/Technologies/Simulation-and-Visualization/Micro-Saint-Sharp - Last visited 28th June 2010.

[71] A. Sweetser. A comparison of system dynamics (sd) and discrete event simulation (des). *Proceedings of the 17th International Conference of The System Dynamics Society*, 1999.

[72] J. R. Swisher, P. D. Hyden, S. H. Jacobson, and L. W. Schruben. Simulation optimization: a survey of simulation optimization techniques and procedures. *WSC '00: Proceedings of the 32nd conference on Winter simulation*, pages 119–128, 2000.

[73] J. R. Swisher, S. H. Jacobson, and E. Yucesan. Discrete-event simulation optimization using ranking, selection, and multiple comparison procedures: A survey. *ACM Transactions on Modeling and Computer Simulation*, 13(2):134–154, April 2003.

[74] XJ Technologies. Anylogic simulation environment. www.xjtek.com - Last visited 28th June 2010.

[75] Telenor. Trygg of effektive mobilbruk. http://www.telenor.com/no/om-oss/var-virksomhet/norden/produkter-og-tjenester/trygg-og-effektiv-mobilbruk - Last visited 28th June 2010.

[76] J. M. Torres, J. M. Sarriegi, J. Santos, and N. Serrano. Managing information systems security: Critical success factors and indicators to measure effectiveness. *LNCS*, 4176:530–545, 2006.

[77] UiO. Translation of the patients' rights act, 1999. http://www.ub.uio.no/ujur/ulovdata/lov-19990702-063-eng.pdf - Last visited 28th June 2010.

[78] András Varga. Omnet++ simulation environment. www.omnetpp.org - Last visited 28th June 2010.

[79] H. Wei, D. Frinke, O. Carter, and C. Ritter. Cost-benefit analysis for network intrusion detection systems. *In Proceedings of the 28th Annual Computer Security Conference*, October 2001.

[80] Wikipedia. Key performance indicator. http://en.wikipedia.org/wiki/Key_performance_indicator - Last visited 28th June 2010.

[81] Wikipedia. Monte carlo method. http://en.wikipedia.org/wiki/Monte_Carlo_method - Last visited 28th June 2010.

[82] Wikipedia. Utility theory. http://en.wikipedia.org/wiki/Utility - Last visited 28th June 2010.

[83] R. K. Yin. *Case Study Research, Design and Methods*. Sage Publications, 3 edition, 2003.

[84] B. P. Zeigler, H. Praehofer, and T. Kim. *Theory of Modeling and Simulation*. Academic Press, 2000.

A Acronyms and Abbreviations

Minimalistic Model Design (MIMD)	Key Performance Indicators (KPIs)
Business Process Modeling (BPM)	Multiple Comparisons with the Best (MCB)
Common Random Numbers (CRN)	Business Process Execution Language (BPEL)
Return on Investment (ROI)	Critical Success Factors (CSFs)
Business Scorecard (BSC)	Business Process Modeling Notation (BPMN)
Unified Modeling Language (UML)	Integration Definition for Function Modeling 0 (IDEF0)
Information Technology (IT)	Advanced Continuous Simulation Language (ACSL)
Intrusion Detection Systems (IDS)	Continuous System Simulation Language (CSSL)
Warm up period (WT)	Discrete Event System Specifications (DEVS)
Discrete Event Simulation (DES)	Response Surface Methodology (RSM)
Asynchronous team (A-Team)	independent and identically distributed (i.i.d)
Multiple Comparison Procedures (MCP)	Validation and Verification (V&V)
Ranking and Selection (R&S)	Information Technology Infrastructure Library (ITIL)
Annual Loss Expectancy (ALE)	Management by Business Objectives (MBO)
Key Goal Indicator (KGI)	Norwegian Center for Information Securing (NorSIS)
Subject Matter Expert (SME)	Personal Identification Number (PIN)
Physical Access Control (PAC)	Time Between Arrival (TBA)
First In First Out (FIFO)	minimum variance unbiased estimator (m. v. u. e.)
Last In First Out (LIFO)	Chief Information Security Officer (CISO)
Akershus University Hospital (Ahus)	Kompetansesenter for IT i helse- og sosialsektoren (KITH)
Human Resource (HR)	Distribuert Informasjons- og Pasientdatasystem i Sykehus (DIPS)
Electronic Health Record (EHR)	Elektronisk Pasient Journal (EPJ)
Denial-of-Service (DoS)	Web Services Choreography Description Language (WS-CDL)
System Dynamics (SD)	Business Process Diagram (BPD)
Environment Specific Distribution (ESD)	Security Related Distribution (SRD)
Business Activity (BA)	Business-to-Security measure-Interface (BSI)
Gjøvik University College (GUC)	Security measure-to-Adversary model-Interface (SAI)
Design Class Diagram (DCD)	

B Translation of Norwegian Health Sector Terms

The specific translations used in this thesis of the Norwegian terms used in the health sector are listed in Table 19. The translations are based on [21] and [54], in addition to the web sites of both the Ministry of Health and Care Services, and the Norwegian Directory of Health.

Table 19: Norwegian to English translation of health sector specific terms used

Akseptkriteriene	Acceptance criterias
Somatisk Sykehus	Somatic hospital
Antall brudd på vurderingsgarantien	Number of assessment warranty breaches
Pasientrettighetsloven	Patients' Rights Act
Gjennomsnittlig ventetid	Average waiting time
Antall fristebrudd	Number of time limit breaches
Liggetid	Length of stay
Elektronisk Pasient Journal	Electronic Health Record
Styringsbrev	Letter of Direction
Styringsvariabler	Direction variables
Oppdragsdokument	Mission document
Helseforetak	Health division
Helse Sør-Øst	South-Eastern Norway Regional Health Authority
Vurderingstid	Assessment time
Elektronisk Pasient Journal (EPJ) system	Electronic Health Record (EHR) systems
Nødrettstilgang	Use of emergency access
Pasientinformasjon ikke tilgjengelig for behandlende personell	Medical record unavailable
Helseinformasjon ikke oppdatert i journalene	Lack of medical record integrity
Pasientinformasjon på avveie	Breach of medical record handling rules
Revmatismesykehuset	Rheumatism Hospital
Helse- og omsorgsdepartementet	Ministry of Health and Care Services
Sosial og helsedirektoratet	Norwegian directory of Health

C Statistical Notations

All variables used in the statistical calculations done throughout the thesis are described in this Appendix.

Variable	Description
θ	Measure of performance
$\hat{\theta}$	The point estimator (a "best guess" for an unknown (fixed or random) population parameter)
α	Probability of falsely rejecting the statistical hypothesis tested
ϵ	Error criterion, used to determine the measure of accuracy
$t_{n-1,1-\alpha/2}$	*t* distribution, with n-1 degrees of freedom
H	The half-length of the confidence interval
n	Number of replications
n_0	Initial number of replications used to estimate n
σ^2	Population variance
σ_0^2	Initial estimation of population variance
σ	Standard deviation
s.e.	Standard error
z_β	Value determined by used the cumulative distribution function and cumulative normal distribution table from [4])
β	Type I error level
$\overline{X}$	Overall sample mean
k	Number of batches
$\overline{Y}_j$	Mean of the *j*th batch
$\overline{Y}$	Overall sample batch mean
$\widehat{\rho_1}$	Sample lag-1 autocorrelation
C	Independence between batch means
θ_i	Mean performance measure of system i (i = 1,2)
θ_{ij}	The mean performance measure of the *j*th observation of system i
d_j	Individual differences between $\theta_1 j$ and $\theta_2 j$
$\overline{d}$	Average difference between the systems
δ	The true difference between the systems

D Simulation Specific Modeling Notations

Based on ExtendSim's modeling blocks, we will in this Appendix give a short explanation of those blocks presented in this thesis.

Item Blocks

The blocks that are used in our model to process or create *Items* are depicted in Figure 20.

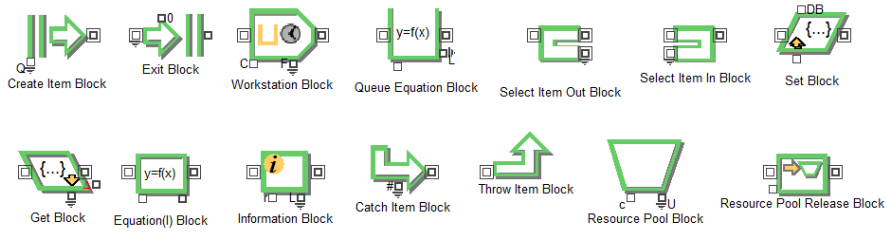

Figure 20: Item blocks used

1. The *Create Item* block generates *Items* either randomly, e.g. with the use of statistical distributions, or it generates *Items* by a predefined schedule. This block can also generate *Values*.
2. The *Exit* block is used when, as the name implies, *Items* exits the model. This can either be when a customer is finished shopping or when a product is shipped out of the factory.
3. The *Workstation* block consists of two blocks in one, a *Queue* and a *Activity* block. The block stores *Items* in its own queue and processes a *Item* when the *Activity* block becomes available and/or some function or condition determines that the *Item* can be processed. A *Workstation* block, or *Activity* block, can represent everything in the real world that performs some type of job or action, e.g. a machine processing a product or a doctor treating a patient.
4. The *Queue Equation* block stores *Items* which are pending to be processed or other types of actions. A *Queue* can for example represent a waiting line for patients. The *Queue* sorts its *Items* either by a First In First Out (FIFO) or a Last In First Out (LIFO) scheme, or it can determine the order of the *Items* by some sort of equation/function.
5. The next two blocks, *Select Item Out* and *Select Item In*, is used respectively to route *Items* to a particular block or to merge *Items* coming from several blocks to one particular block.
6. The *Set* block and *Get* block are used respectively to set attributes of a bypassing *Item* or to read the attributes of a bypassing *Item*. The attributes can either be specified within the block itself, or determined by a value block or a database.

7. The *Equation(I)* block is perhaps one of the most versatile blocks found in the library. It can for example output a particular value based on a specific attribute found within the *Item* or read/write information to a database. The *Equation(I)* block can basically, with the use of the *ModL* language, replicate most of the functions found in the other blocks.
8. The *Catch Item* and *Throw Item* blocks allow us to send a particular item from one place in the model to another without directly linking these two points.
9. The *Resource Pool* block stores items possible to utilize in the model, e.g. doctors, while the *Resource Pool Release* block, as the name implies, releases items taken from the *pool*. That is, the items become available.

Value Blocks

Blocks that are used in our model to process or create *Values* are depicted in Figure 21.

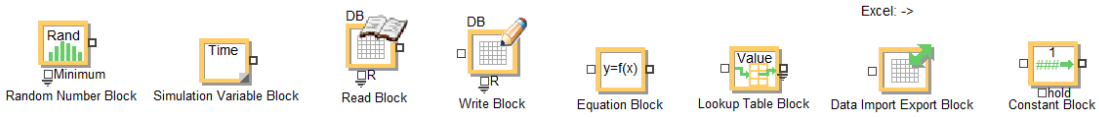

Figure 21: Value blocks used

1. The *Random Number* block, generates, as the name implies, random *Values* based on the minimum standard random number generator.
2. The *Simulation Variable* block outputs the value of a simulation variable. To be more specific, it outputs the value of either the *current run number, current step, current time, end time, number of runs, number of steps, start time, time step,* or a *random seed*.
3. The *Read* and *Write* block is used to read and write data from and to a database. The data which is read from the database is used as input to an *Items* attribute or in order to change the behavior of the model during simulation. Likewise, data found within items can be updated to the database with the *Write* block.
4. The *Equation* block is very similar to the aforementioned *Equation(I)* block. The difference between the two is that this block only allows you to enter formulas and equations to calculate values for models, whereas the *Equation(I)* block also allows us to change the *Items* directly. That said, the *Equation* block is a powerful block which have many applications.
5. The fifth block, *Lookup Table*, is in our model, amongst other things, used to determine the state transition of an *Item*. It does this by reading a specific attribute of the *Item* by utilizing the *Get* block, and then looks up which value it should change the attribute to. The corresponding value is then sent to the *Set* block which alters the attribute of the *Item*. The *Lookup Table* block has many other applications, e.g. determining the schedule which again determines the behavior of the model.
6. The second to last value block that we have applied in our model is the *Data Import Export* block. This block allows us to generate a report that lists the context of the database

which includes all input data and values of our KPIs. ExtendSim has also the possibility of generating a report on the blocks them self, but we are able to generate a more customized report by utilizing this block.

7. The last value block, is the *Constant* block, which, as the name implies, provides a constant value to the model.

Miscellaneous

In addition to the blocks already mentioned, one can, when wanting to creating a hierarchical model, gather several different blocks and create one *Hierarchical block* as depicted in Figure 22. It is beneficial to gather several blocks like this when we want to reduce the number of block visible for the user at a particular level. Should the user want or need to look closer at the interconnection between the blocks within the Hierarchical block, the user can simply double click on the block in order to go one level down in the hierarchy. The *Hierarchical block* has input and output connectors, where the number of the different connectors is determined by the blocks within the *Hierarchical block* itself.

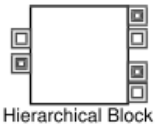

Figure 22: Hierarchical block

Almost all blocks[1] are connected with *Lines* which transfers a *Item* or *Value* from one point to another. By default, the flow of *Items* and *Values* goes from left to right, but where this is not the case, an arrow can be added to specify the direction of the flow. Figure 23 and Figure 24 illustrates both these situations, where the *Items* in the Figure 23 flows from left to right and an arrow is therefore left out. In the example found in Figure 24 however, the lines are, due to practical reasons, structured in such a way that the *Items* flow in both directions and a clarification is hence needed.

Figure 23: Examples of lines connecting blocks

[1] Some blocks, e.g. the *Data Import Export* block, are not directly connected to the rest of the blocks.

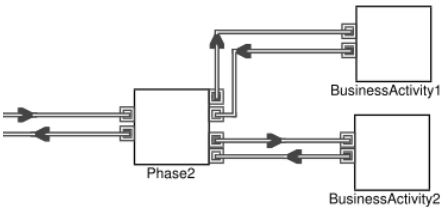

Figure 24: Examples of lines with arrows connecting blocks

E Simulation Background

We will in this Appendix presents a short introduction into the area of simulation. The content found in this Appendix is, unless otherwise specified, based on the work of Banks et al. ([4]). All variables used in equations presented in this Appendix are explained in Appendix C.

A model can be defined as a representation of a system for the purpose of studying the system, while simulation can be defined as the imitation of the operation of a real-world process or system over time. It is normal to differentiate between two types of systems, namely terminating and non-terminating systems, where the main difference between them lies in the time period each system provides useful data.

A terminating system has an obvious time when no more useful information will be obtained by running the simulation longer. Examples of such systems are service center that is only open a specific number of hours each day, e.g. 10, where we then also sets the simulation end time to 10 hours. Such systems do not typically reach a continuing steady state, and the purpose in modeling them is usually to look for changes and identify trends. Since the initial conditions in terminating systems will have an impact on results, it is important that they be both realistic and representative of the actual system [35]. A non-terminating system is a system that runs continuously or at least over a very long period of time, e.g. hospital emergency rooms. A simulation of a non-terminating system starts at simulation time 0 under initial conditions defined by the analyst and runs for some analyst-specified period of time T_E. Usually, the analyst wants to study steady-state, or long-run, properties of the system, i.e. properties that are not influenced by the initial conditions of the model at time 0. T_E is not determined by the nature of the problem, but rather determined by the analyst as one parameter in the design of the simulation experiment. Possible objectives or goals with steady-state simulations can be to estimate production efficiencies, the long-run average throughput and utilization of each computer, or the cycle time for parts, i.e. the time from release into the factory until completion, or the time to fulfill a customer's request.

As mentioned, a model is defined as a representation of a system for the purpose of studying the system. A model can further be classified as being mathematical or physical. A mathematical model uses symbolic notation and mathematical equations to represent a system, and a simulation model is a particular type of mathematical model of a system. Simulation models may be further classified as being static or dynamic, deterministic or stochastic, and discrete or continuous.

A static simulation model represents a system at a particular point in time, while dynamic simulation models represent systems as they change over time. Simulation models are analyzed by numerical methods rather than by analytical methods. Analytical methods employ the deductive reasoning of mathematics to "solve" the model. Numerical methods employ computational procedures to "solve" mathematical models.

Simulation models that contain no random variables are classified as deterministic. Such models have a known set of inputs, which will result in a unique set of outputs. A stochastic

simulation model has one or more random variables as inputs. Random inputs lead to random outputs, and because of this, the output can only be considered as estimates of the true characteristics of a model.

When simulating, we use statistical distributions in order to represent real world data. Several types of such distributions exist, including exponential, normal, Poisson and triangular distributions. Based on the description found in [35], we present a short description of these distributions. Normal distributions are most often used when events are due to natural rather than man-made causes, while exponential and Poisson distributions are primarily used to define intervals between occurrences. The triangular distributions are used for activity times where only three pieces of information (min, max and most likely values) are known, and they are usually more appropriate for business processes than the uniform distribution since it provides a good first approximation of the true values

Due to the inherent randomness found within stochastic models, applying methods designed to increase the confidence in the generated results is required. Usually, one achieves this by either performing several simulation replications or by running one (long) replication, where the choice of method usually depends on whether the system is terminating or non-terminating[1].

It is important to notice that a simulation replication and a simulation run are not the same. A simulation run consists of the complete process, from pressing the "run button" until the results is outputted by the simulation environment. A replication on the other hand consists of what happens within the model from simulation time 0 until the stopping time T_E or stopping event E becomes true. Hence, several replications can be found within one simulation run. How we determine the number of replications and the length of each of these, is done differently depending on the type of system, and can also be done in several ways for each type of system.

When determining when the simulation has reached steady state, two main approaches can be used, either ensemble average or cumulative average. The ensemble average is the sample mean of independent and identically distributed (i.i.d.)[2] observations, a confidence interval based on the t distribution can be placed around each point[3]. This is the preferred method to determine a deletion point. Ensemble averages can be smoothed further by plotting a moving average, rather than the original ensemble averages. In a moving average, each plotted point is actually the average of several adjacent ensemble averages. Cumulative average sample mean becomes less variable as more data are averaged. The left side of the curve will always be less smooth than the right side. Because cumulative averages contain all observations, it tends to converge more slowly to long-run performance than ensemble averages. Cumulative averages should be used only if it is not feasible to compute ensemble averages, such as when only a single replication is possible.

As mentioned, determining when a simulation stabilizes is important in steady-state simulation. However, since we are often interested in several different output performance measures at once, which could approach steady state at different rates, it is important to examine each performance measure individually and use a deletion point that is adequate for all of them.

[1] Also known as steady-state system

[2] Independent means that the data is based on different random numbers, while identically distributed means that one is running the same model on each replication

[3] We can then judge whether or not the plot is precise enough to judge that bias has diminished

Should the output data not be statistical independent, i.e. they are autocorrelated, additional treatment is needed before we can determine the systems performance measure. Autocorrelated sequence of data, sometime called a time series, almost always is the case when the output is a sequence of observations from within a single replication. The covariance between two random variables in the time series depends only on the number of observations between them, called the *lag*. The lag-*k* autocorrelation is the correlation between any two observations *k* apart.

When the lag-k autocorrelation is larger than zero ($\widehat{\rho_k} > 0$) for all k (or most k), the time series is said to be positively autocorrelated. In this case, large observations tend to be followed by large observations, small observations by small ones. Such a series will tend to drift slowly above and then below its mean. The output data from most queuing simulations are positively autocorrelated. If some of the $\widehat{\rho_k} < 0$, the output will display the characteristics of negative autocorrelation. In this case, large observations tend to be followed by small observations, and vice versa. The output of certain inventory simulations might be negatively autocorrelated.

If the autocorrelations $\widehat{\rho_k}$ are primarily positive, and this correlation were ignored, the simulation analyst would have unjustified confidence in the apparent precision of the point estimator due to the shortness of the confidence interval. When positive autocorrelation is present in the output data, the true variance of the point estimator, $\overline{\theta}$, can be many times greater than is indicated by σ^2/n. If the autocorrelations $\widehat{\rho_k}$ are substantially negative the true precision of the point estimator $\overline{\theta}$ would be greater than what is indicated by its variance estimator σ^2/n. This error is less serious than the case above, because we are unlikely to make incorrect decisions if our estimate is actually more precise than we think it is.

In order to eliminate or at least reduce the effects of autocorrelation in non-terminating systems, two methods can be used, namely the replication method or the batch means method. When using the replication method (independent replication), due to the initialization bias found in non-terminating systems, data must be deleted in each replication. Since bias is not affected by the number of replications, in order to reduce the bias (increase accuracy), more data is needed to be deleted within each replication, and each replications length will be needed to be extended in order to collected the same amount of observations within each replication. One disadvantage of the replication method is that data must be deleted on each replication. The deleted data are then wasted data, or at least lost information. This suggests that there might be merit in using an experiment design that is based on a single, long replication. The disadvantage of a single-replication design arises when we try to compute the standard error of the sample means. Since we only have data from within one replication, the data are dependent, and the usual estimator i biased. The method of batch means attempts to solve this problem by divide the output data from one replication (after appropriate deletion) into a few large batches and then treating the means of these batches as if they were independent.

When simulating stochastic models, the use of random numbers is as mentioned an essential element. However, how one utilizes the random number streams in a model when simulating becomes an important issue when several systems designs are to be compared. Two different approaches can be used, namely independent sampling and correlated sampling. Correlated sampling is also known as Common Random Numbers (CRN). Independent sampling means that different and independent random number streams will be used to simulate the two systems. All

observations of simulated system 1 are statistically independent of all the observations of simulated system 2. With the CRN implementation, each system, as the name implies, is compared with the use of the same random numbers. One benefit of using CRN, is that it provides us with greater precision in the estimation of θ_1 - θ_2, compared to *Independent sampling*, provided that the number of replications and length of simulation is the same.

F Determining Simulation Setup for Terminating Systems

The methods found in this Appendix are based on those described in [11] and [4]. All variables used in the equations presented in this Appendix are explained in Appendix C.

In the case of terminating systems, when determining the simulation setup it is necessary to determine two things, namely the sample size and the simulation length. Since the simulation length typically is established by the context of the problem, no additional description of determining this is needed. We hence only focus on determining the sample size, i.e. determine how many replications are needed.

In order to answer the question of "how many replications do we need to make?", we apply the *method of independent replications*. The method includes the following steps [4]:

1. Establish the measure of performance, θ, for the analysis, e.g. the mean time in the system. The selection of the measure of performance is based on the KPIs of interest.
2. Decide the type of confidence and accuracy[1] that we seek, i.e. α and ϵ respectively
3. Run the model for a small number of replications, n_0, e.g. between 5 and 10 replications
4. Obtain an initial estimate σ_0^2 of the population variance σ^2
5. Determine the initial estimate for n, given by

$$n \geq \left(\frac{z_{\alpha/2}\sigma_0}{\epsilon} \right)^2 \tag{F.1}$$

6. Determine the smallest integer of n satisfying $n \geq n_0$ and

$$n \geq \left(\frac{z_{\alpha/2,n-1}\sigma_0}{\epsilon} \right)^2 \tag{F.2}$$

7. Make n number of replications
8. Based on the overall sample mean, $\overline{X}$, of the replications, compute the (1-α)% confidence interval based on Equation F.3.

$$\overline{X} - t_{n-1,1-\alpha/2}\frac{\sigma}{\sqrt{n}} \leq \theta \leq \overline{X} + t_{n-1,1-\alpha/2}\frac{\sigma}{\sqrt{n}} \tag{F.3}$$

If the value of H, the half-length of the confidence interval, is approximately ϵ or smaller, then n is a sufficient number of replications and we have determined the sample size, and hence also the simulation setup. However, should the confidence interval be too large, i.e. H is larger than ϵ, the procedure must be repeated in order to determine a larger n satisfying the demand $H \leq \epsilon$.

[1] Error criterion (ϵ) can only be specified if the sample size can be increased. It is desired to estimate θ by $\overline{x}$ to within $\pm\epsilon$ with high probability, say at least 1-α.

G Collected Data

We will in this Appendix present the collected data used as a basis for our input data. In addition to presenting the actual data, we will also describe how the data have been collected and how the collected data have been processed. In order to utilize the data, the processing becomes important since several different sources have been used and the data is hence tainted by these sources.

The majority of the data have been collected from Ahus, Rheumatism Hospital and Ergo Group[1]. In addition, in order to gain security related data, statistics and findings from several different sources have been utilized, including [66], [75], [65], [48], [62], [24], [27], and [26].

The collected data, unless otherwise specified, is rescaled based on Equation G.1 such that we get a uniform value to be used in the triangular distributions, which again determines the behavior of the model. By rescaling the collected data, we mitigate the problem of having collected data from sources with a different size of the user population. Furthermore, by rescaling the data to a "per user" scale, others are able to easily adapt the data to their particular organization.

$$\left(\frac{\texttt{Number of observations}}{\texttt{Minutes} \times \texttt{Number of employees}}\right)^{-1} \tag{G.1}$$

G.1 Business activity related data

In this section we will present the data collect at the Rheumatism Hospital with regards to the duration of each business activity identified. The data, which are all based on SME opinions, is listed in Table 20.

Initially, all data found in Table 20 was to be used, however, due to alterations of our methodology, only the "Treatment Duration" data was in fact used. We have however utilized some of the data in order to make assumptions with regards to the business activity "Approve Documents", in which we did not get any data on. That is, the figures used in our model with regards to the "Approve Document Duration" are based on the rest of the data collected by the Rheumatism Hospital.

As only a minimum and maximum, or a mean estimated duration was provided by the Rheumatism Hospital, the remaining data, i.e. either the mean time based on the provided minimum and maximum value or the minimum and maximum value based on the provided mean value, have been estimated by the author. These estimates are listed in *italic* in Table 20. We needed to estimate the remaining data in order to get all three values for the triangular distribution. Ideally, one should of course use only provided data. However, as the estimates are closely linked to the provided data, the effect of these decisions should be minimal.

Since the collected data is not affected by the number of employees the data have not been rescaled.

[1] Ergo Group is a leading Nordic IT company

Table 20: Data from Rheumatism Hospital

Business Activity	**Duration of Business Activity**		
	Minimum	Maximum	Mean
Request assessment duration	3 min	30 min	*16 min*
Reject request duration	1 min	10 min	*5 min*
Admission processing duration	*1 min*	*2 min*	1 min
Patient admission duration	*25 min*	*35 min*	30 min
Write entry duration	7 min	25 min	*16 min*
Patient treatment duration	*50 min*	*70 min*	60 min
Patient discharge duration	*1 min*	*2 min*	1 min

G.2 Authentication related data

We will in this Section present the data received from Ahus and Ergo Group which is used to determine the number of authentication failure occurrences, as well as determining the duration of each occurrence.

G.2.1 Number of smart card related requests

Based on information received from Ahus, the number of requests to the ID office each month by employees at Ahus has been determined and is listed in Table 21. These figures are based on Ahus' second year of using the ID office, and the numbers from 2008 are left out as these will not represent a normal year due to the introduction of the system. We do however assume that the numbers from 2009 represent a normal year of requests from employees.

In Table 21, the second column called "*New cards*", represents a request from a new employee. When an employee loses his/her card, a "*Replacement card*" request is placed while a "*Temporary card*" request is placed when the user only has forgotten his/her card at home or similar. Although not relevant for our model, a "temporary card" has a validity of 48 hours. The last column, "*New PIN code*", consists of requests from users who have forgotten their PIN code and needs to reset this.

The "Average" month which we have created, includes the lowest and highest number of requests during a year, as well as the mean (most likely) number of requests during a year. We have created this "Average" month in order to be able to use the requests collected during one year, as the input in our triangular distribution.

Since the data received from Ahus is based on 6000 employees and presented in "requests per month", we rescale the data based on Equation G.1 such that we are able to use the data in our model.

To illustrate, the rescaling of the mean number of "*new PIN code requests*" in the "Average" month becomes $(213/(43200*6000))^{-1} = 1216901$, i.e. the time between arrival of forgotten PIN code per user is 1216901 minutes. When utilized in our model, we will scale these figures to fit our scenario, i.e. we divide the TBA with 12, which yields a TBA of 101408 minutes.

All figures found in Table 21 have been rescaled and the processed figures can be found in Table 22. It is important to note that the rows "Lowest" and "Highest" in Table 21 represents the rows "Max" and "Min" respectively in Table 22, i.e. the higher the number of requests the lower the Time Between Arrival (TBA).

Table 21: Data from ID Office at Ahus (2009)

	Number of requests			
Month	**New cards**	**Replacement cards**	**Temporary cards**	**New PIN codes**
Jan	217	77	79	322
Feb	321	54	76	228
Mar	239	73	116	251
Apr	176	66	113	208
May	190	55	117	210
June	339	84	178	248
July	186	89	103	155
Aug	249	72	100	220
Sept	214	98	136	196
Oct	171	86	124	201
Nov	118	58	104	188
Dec	52	61	112	129
Total	**2472**	**873**	**1358**	**2556**
"Average" month				
Lowest	52	54	76	129
Highest	339	98	178	322
Mean	206	73	113	213

Table 22: Distribution used for smart card related requests

Request type	**Time Between Arrivals (TBA) of requests**		
	Minimum	**Maximum**	**Most likely**
New card request (/user)	764602 min	4984615 min	1258252 min
Replacement card request (/user)	2644898 min	4800000 min	3550685 min
Temporary card request (/user)	1456180 min	3410526 min	2293805 min
New PIN code request (/user)	804969 min	2009302 min	1216901 min

We make a simplification with regards to how often a doctor forgets a PIN code or loses the smart card. That is, we do not include issues like when a doctor is more likely to forget a PIN code or forget the smart card, e.g. after weekends and after holiday for instance. However, our input data surrounding the smart card product should be a good estimate of an average month given that it is based on data from one year.

G.2.2 Number of password related requests

We where unfortunately not able to collect the number of new password requests from Ahus, and we instead chose to collect these figures from Ergo Group based on two reasons. First of all, Ergo Group have a user population which, although smaller than Ahus', is still of considerable size. Secondly, we had access to such data. The collected data can be seen in Table 23.

The number of users which the data collected from Ergo Group is based on is different from

Table 23: Data from Ergo Group (week 11)

	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Number of password requests	18	6	8	12	12	4	4

that of Ahus', namely 2400 users as opposed to Ahus' 6000. This is however handled by Equation G.1 and this should therefore not affect the validity of the data. There are however a couple of issues surrounding these figures which requires us to make some assumptions. The data is first of all not based on the same users or the same business sector. In order to be able to use the data, we hence need to make the assumption that a person, regardless of the type of business sector, forgets a password with an equal probability. The second issue is concerned with the fact that the data was only possible to be collected during a very short period of time, more exactly one week. Because of this, we need to make another assumption with regards to the data. We will assume that the week which the data have been collected (week 11) represents an average week of an average month. The fact that the data is collected during a period where no holidays or other inconsistencies in a users daily activities occur, strengthen our belief in that this assumption holds. In other words, the users have recently used their password and the number of requests should therefore not be affect by abnormal activities.

Provided that our assumptions hold, we can use this data in our model. We first need to determine how the collected data represents an "average" month instead of a week. Since we cannot simply multiply our sample week with four in order to get a month's worth of data, as this will only give us 28 days in total, we need to add two additional days in order to get data for a month[2]. By using the 64 requests which occur during a week as our baseline, we multiply this baseline by four and use the lowest, highest and average number of request during a week as the two additional days to complete our data sample, which then also provides us with a triangular distribution. That is, during an "average" month we will observe between 264 ($64 \times 4 + (4 + 4)$) and 292 ($64 \times 4 + (18 + 18)$) requests, where 274 ($64 \times 4 + (9 + 9)$) requests are most likely. The figures listed in Table 24 have been calculated based on applying the "average" month figures in Equation G.1.

Similar to the smart card data, we make simplifications with regards to how often a doctor forgets a password. That is, we do not include issues like when a doctor is more likely to forget a password. However, unlike the smart card data, we have a small basis for our data, so we cannot with an equally large degree of confidence state that our figures include such real life issues.

Table 24: Distribution for password resetting

	Time Between Arrivals (TBA) of requests		
	Minimum	**Maximum**	**Most likely**
New password request (/user)	355068 min	392727 min	378394 min

G.2.3 Duration of each request type

In addition to determining how many requests there are during one month, we also need to determine the duration of each request. That is, how long does it take for a doctor to reset his/her PIN code or password, and how long does it take to get a new smart card. The data used in our model, which is listed in Table 25, have been determined based on the Subject Matter Expert (SME) opinions of Ellef Mørk and Halvor Sandodden at Ahus. As with the business activity duration, the data does not need to be rescaled since the collected data is not affected by the

[2]Recall that we define a month as 30 days

Table 25: Distribution for authentication failure duration

	Duration of processing new requests		
	Minimum	**Maximum**	**Most likely**
New password duration	3 min	10 min	5 min
New PIN duration	6 min	17 min	11 min
New PIN @ PAC duration	1 min	3 min	1 min
New card duration	10 min	21 min	15 min
New card @ PAC duration	3 min	5 min	4 min

number of employees.

Because passwords can be reset by making a phone call to the ID office, the time is about half that of resetting the PIN code or getting a new smart card. In the case of using smart cards as the security measure, the employees needs to physically go to the ID office, which may be up to a 5 minutes' walk from where he/she perform the business activity. Assuming that the employee needs to go back to where he/she came from, the maximum time used on resetting a PIN code or getting a new card becomes 10 minutes plus the time it takes actually takes to resetting the PIN code or creating a new card. When the authentication failure occur at the physical access control (PAC), i.e. when a employee arrives at work, the total duration includes only a minor extra time delay in addition to the time it takes to actually reset the PIN or creating a new card. Because there is such a difference, we have also included two different distributions.

We have simplified the data by not taken into account that the duration changes between "office hours" and outside "office hours". However, as this affects both types of security measures equally, this will not affect the outcome. However, what should have been included is the fact that when a doctor is located at a distributed location where a ID office is not established, there is a considerable increase in the duration should a doctor forgets his PIN code or needs a new card. If such cases would have been included, in a worst case scenario, an additional hour could have been added in the duration.

G.3 Security threat related data

We will in this section present the collected data which are related to security threats. We have chosen to divide the data into two categories, namely snooping and unauthorized access. Where we have access to SME opinions in the case of snooping, no such data was possible to collect with regards to unauthorized access data.

G.3.1 Snooping

Based on the SME opinions of Ellef Mørk at Ahus, there are about 10 cases of snooping each year at Ahus. Assuming that there at most is 12 cases, and at least 8 cases of snooping during a year, by utilize Equation G.1, we get the figures listed in Table 26.

Table 26: Distribution of snooping

	Time Between Arrivals (TBA)		
	Minimum	**Maximum**	**Most likely**
Snooping (/user)	259200000 min	388800000 min	311040000 min

We assume that these figures stay constant regardless of which security measure is selected. The justification for this assumption is that all employees are fully aware of the consequences should they be caught, and it is further justified by the fact that both types of authentication mechanisms will log who accessed the files, i.e. in both cases, the employee knows that the actions will be possible to traced. However, should an employee chose do perform snooping, we assume that such actions will succeed, i.e. there is a 100 % success rate.

G.3.2 Targeted attacks

This type of data is difficult to collect in general due to several reasons, including the fact that few are willing to share such information and that the number of incidents most likely will be underreported. Because of this it also becomes very difficult to get good figures from the same sources as we have used so far. Because of this we choose to use generalized statistics from several different sources. This will however reduce the applicability of the data, and we will also get a higher degree of uncertainty concerning the accuracy of the data.

Recalling that for a targeted unauthorized attack to be successful, the attack needs to be performed from within. That is, the attacker needs physical access in order to be able to apply the attack. This makes most of the available data useless as these statistics are based on remote, automated attacks. Because of this, estimating the number of attempts at unauthorized access (not including snooping), and the success rate of these attacks, is difficult. However, we have based our figures on a set of different sources ([66], [75], [65], [48], [62], [24], [27], and [26]) which hopefully provides us with increased confidence in the figures used.

Since there are no specific figures on the number of such attacks, we need to draw comparison from other types of attacks. Based on the figures found in [66], we notice that about 23% of the organizations in the survey have experienced targeted malicious attacks, as opposed to general virus attacks. The number of targeted malicious attacks ranges from 1 to 5 attacks during a year, and although these figures are not directly applicable, we still choose to apply these figures as a "worst case" situation. The figures used to determine the time between arrivals of attacks are listed in Table 27, where the figures are based on a minimum and maximum number of attacks during a year being 1 and 5 respectively, and where we assume that in average, 2 attacks occur.

Table 27: Distribution for unauthorized access attack

	Time Between Arrivals (TBA)		
	Minimum	**Maximum**	**Most likely**
Unauthorized access attempts	103680 min	518400 min	259200 min

The success rate of a social engineering attack ranges from 20 % to 100 % ([24], [62] and [48]). However, when assuming that the social engineering attacks are targeted, and not general phishing attacks, we assume that an attack has a 50 % success rate. However, because the attacker needs to identify himself to the ID office in the case of using smart cards, the success rate is reduced to 20 % for resetting the PIN code, while even lower (10 %) for creating a new card.

Based on general figures from [75] and [65], we know that about 100000 mobile phones are lost or stolen each year, and that in total there is about 5250000 mobile phones in Norway. If we assume a worst case situation where all phones are lost also is found by an adversary, and

that the employees at a hospital represents the Norwegian population in general, then about 2 % of the mobile phones are stolen from the doctors each year. Hence, we assume that there is a 2 % probability of successfully stealing a phone. Although this is a quite modest estimate, since it is likely to assume that an adversary who wants a particular mobile phone most likely will succeed in stealing it, we need to take into considerations other elements such as the fact that most people rarely leave their phone unattended. Since the figures used to estimate the success rate of mobile phones are somewhat vague, and because employees are equally likely to watch over their smart cards as they are with their mobile phone, we chose to use the same figure for the probability of stealing a smart card.

To summarize, the figures used for determining the success rate when simulating unauthorized access attacks are listed in Table 28.

Table 28: Success rate of unauthorized access attack

Attack type	**Success rate**
Get password/PIN from doctor	50 %
Reset password	50 %
Reset PIN	20 %
Create new card	10 %
Theft of mobile	2 %
Theft of card	2 %

It should be noted that all data presented in this Section, with the exception of snooping which we have good data on, are not gathered from our main sources, and the quality of the data with regards to the validity will be lower. The fact that it is so difficult to get hold of relevant and useful data is a huge problem, one which we discuss in more details in Chapter 11.

H Flow Charts

This appendix includes the flow charts that we have received from the Rheumatism Hospital (Figure 25 to 32) and those flow charts that have been developed based on collaboration with Buypass (Figure 33 to 39).

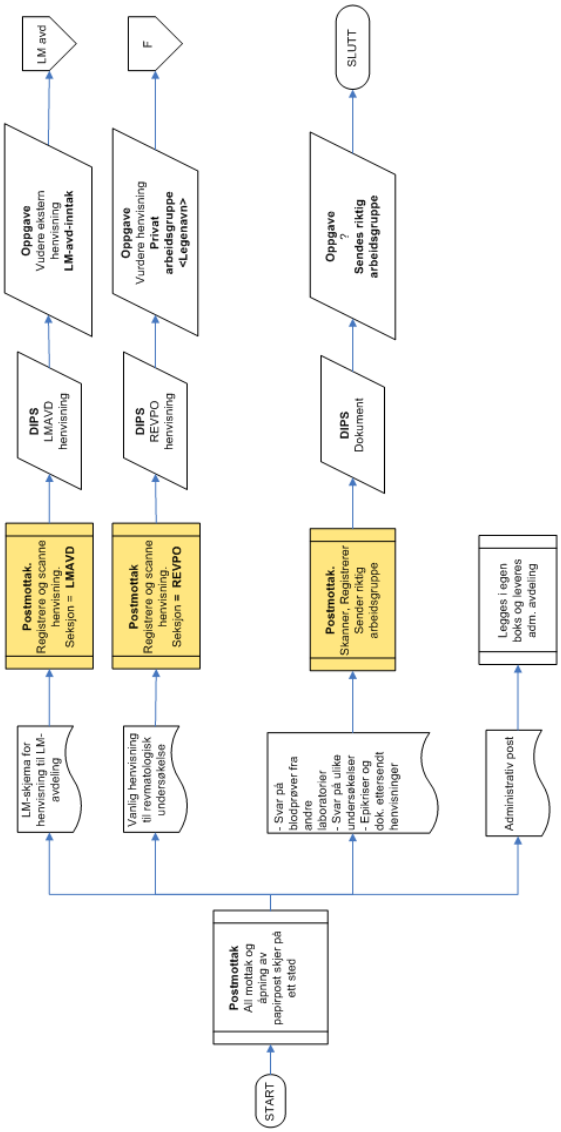

Figure 25: Post office: Arrival of request

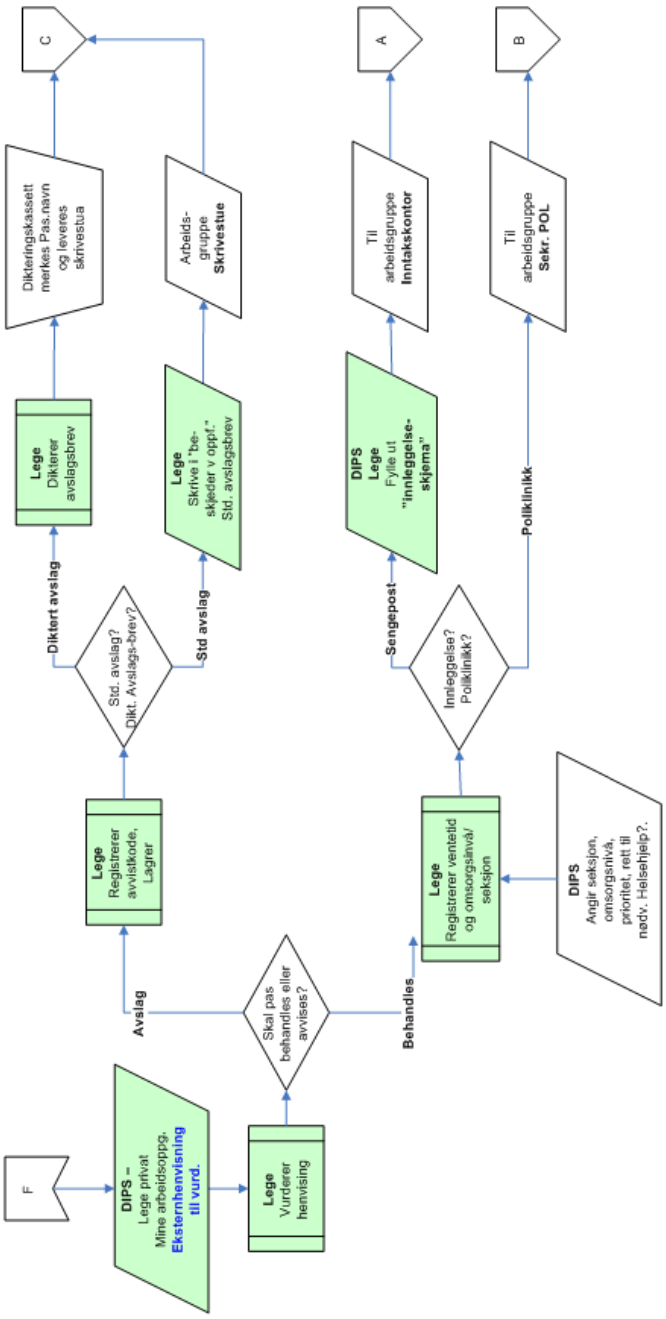

Figure 26: Request assessment performed by doctor

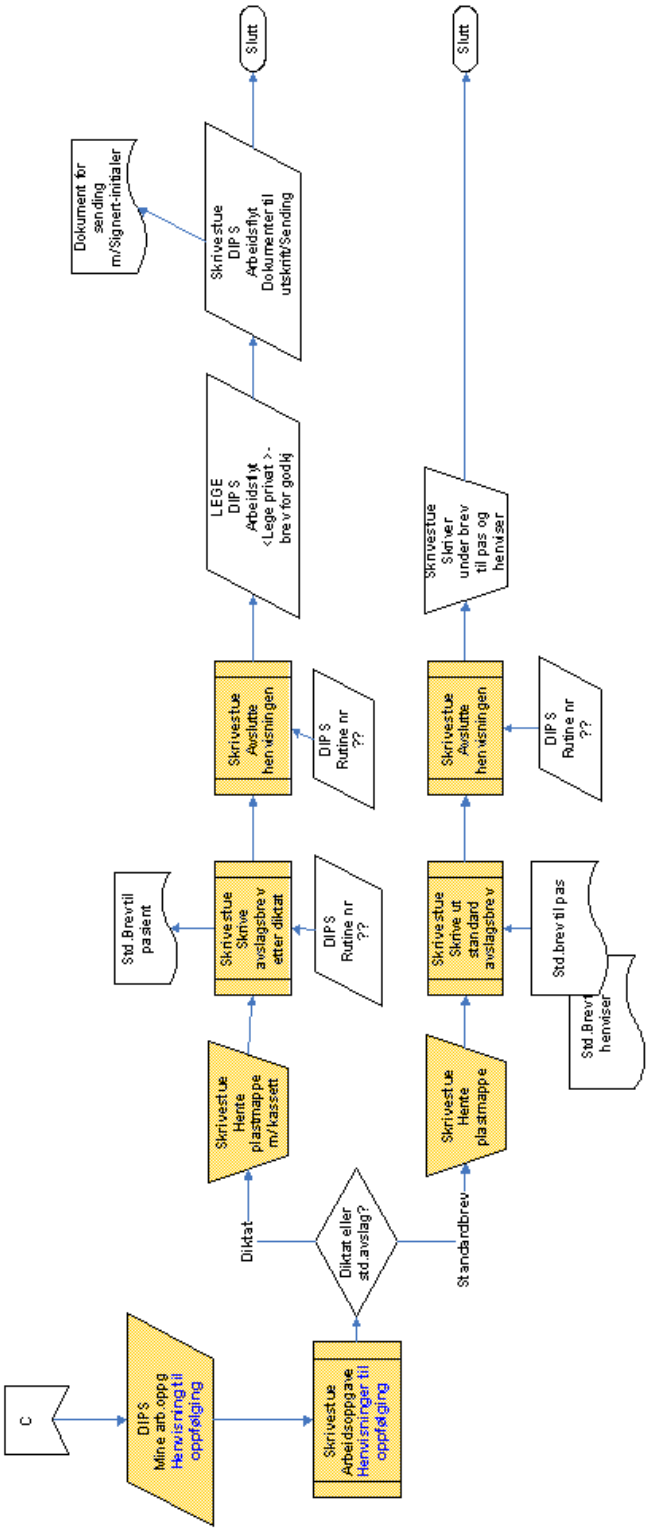

Figure 27: Writing and sending rejected request

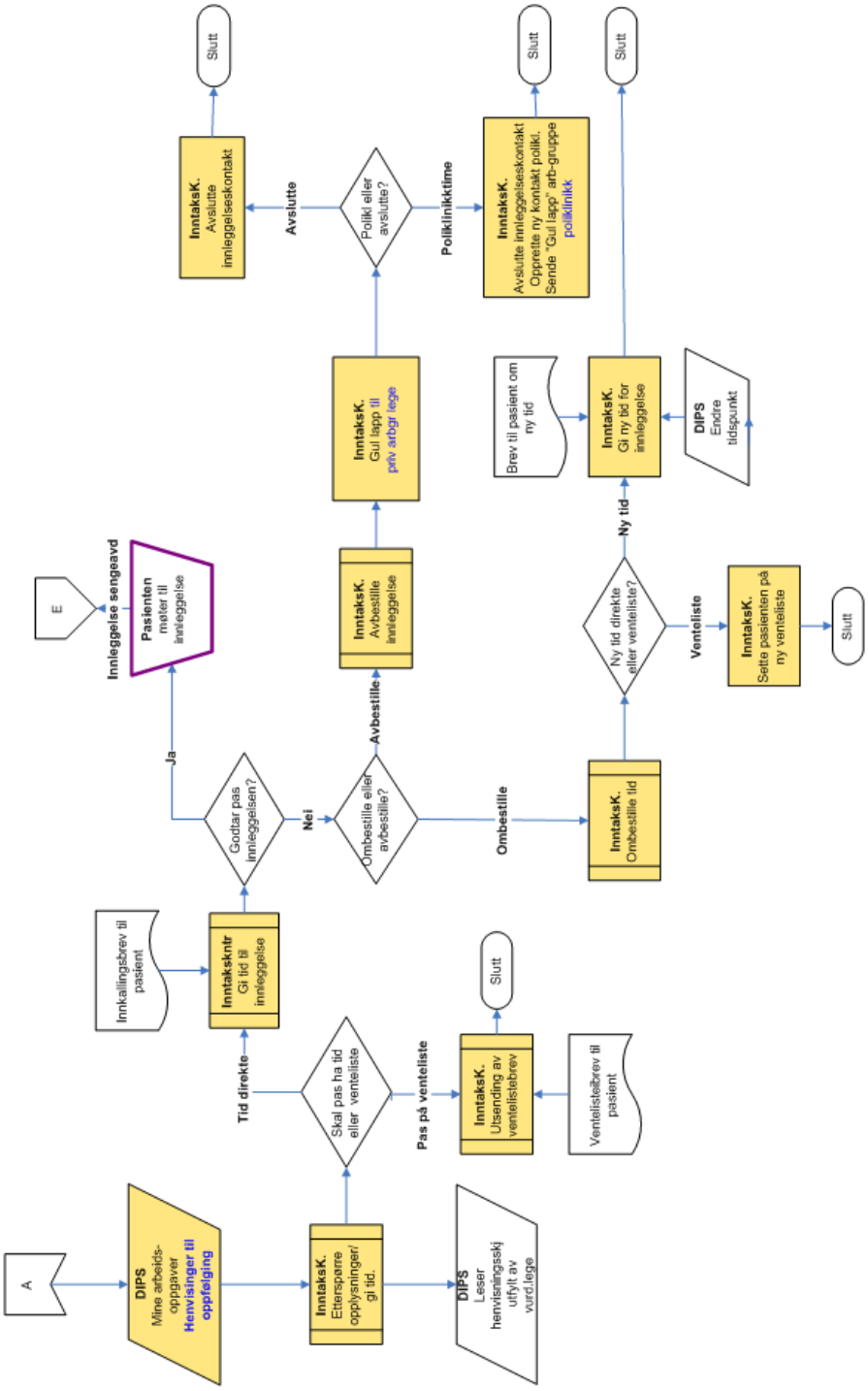

Figure 28: Prepare for admission of patient

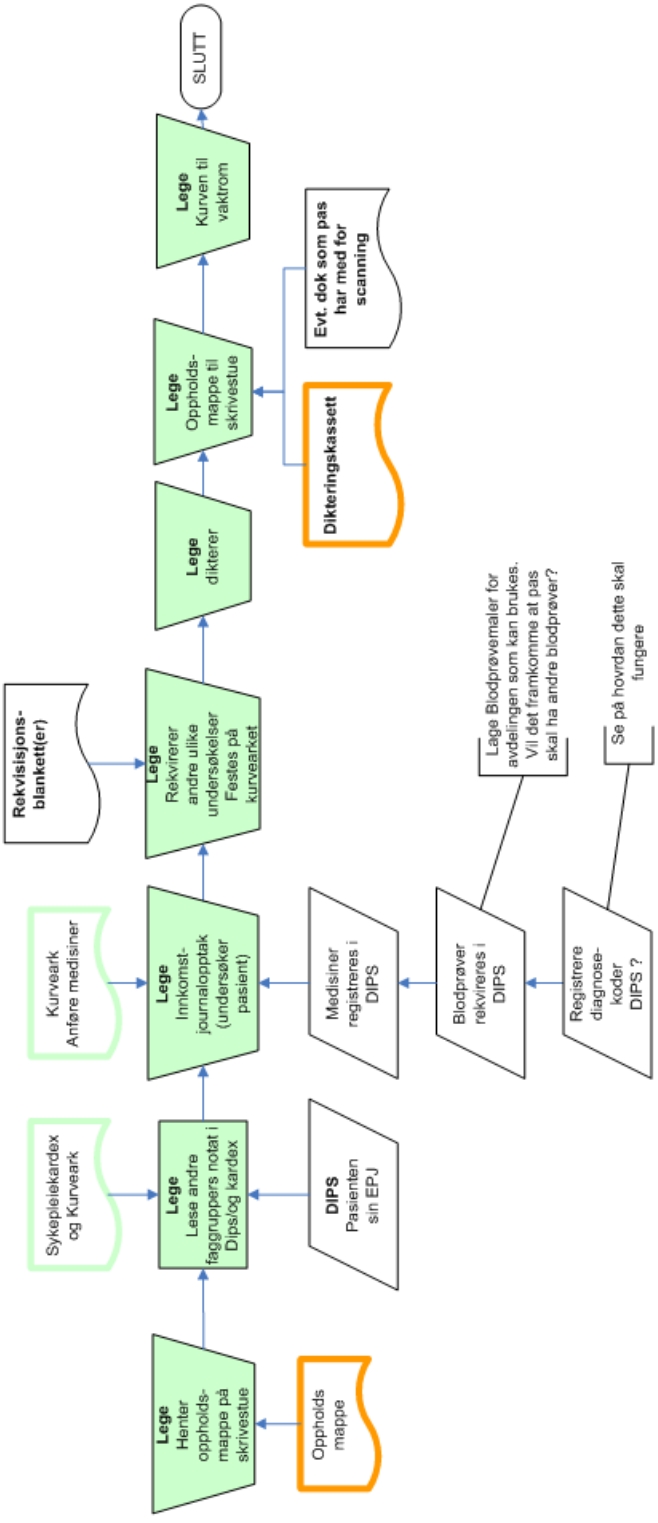

Figure 29: Admission of patient with creation of journal

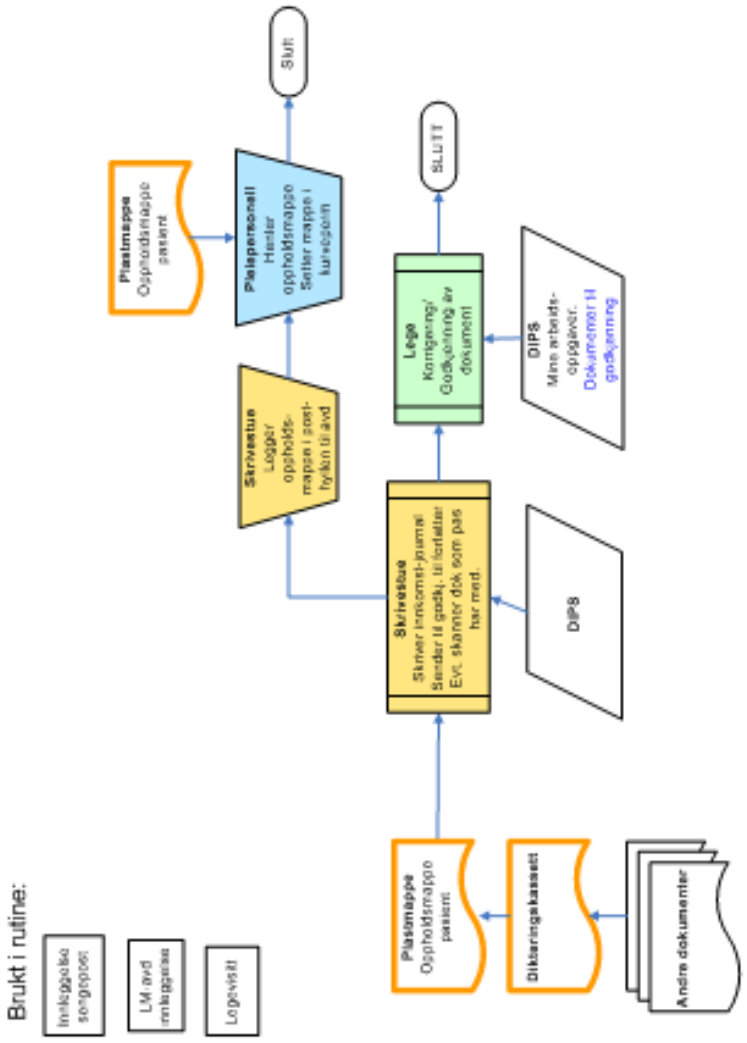

Figure 30: Writing the admission journal

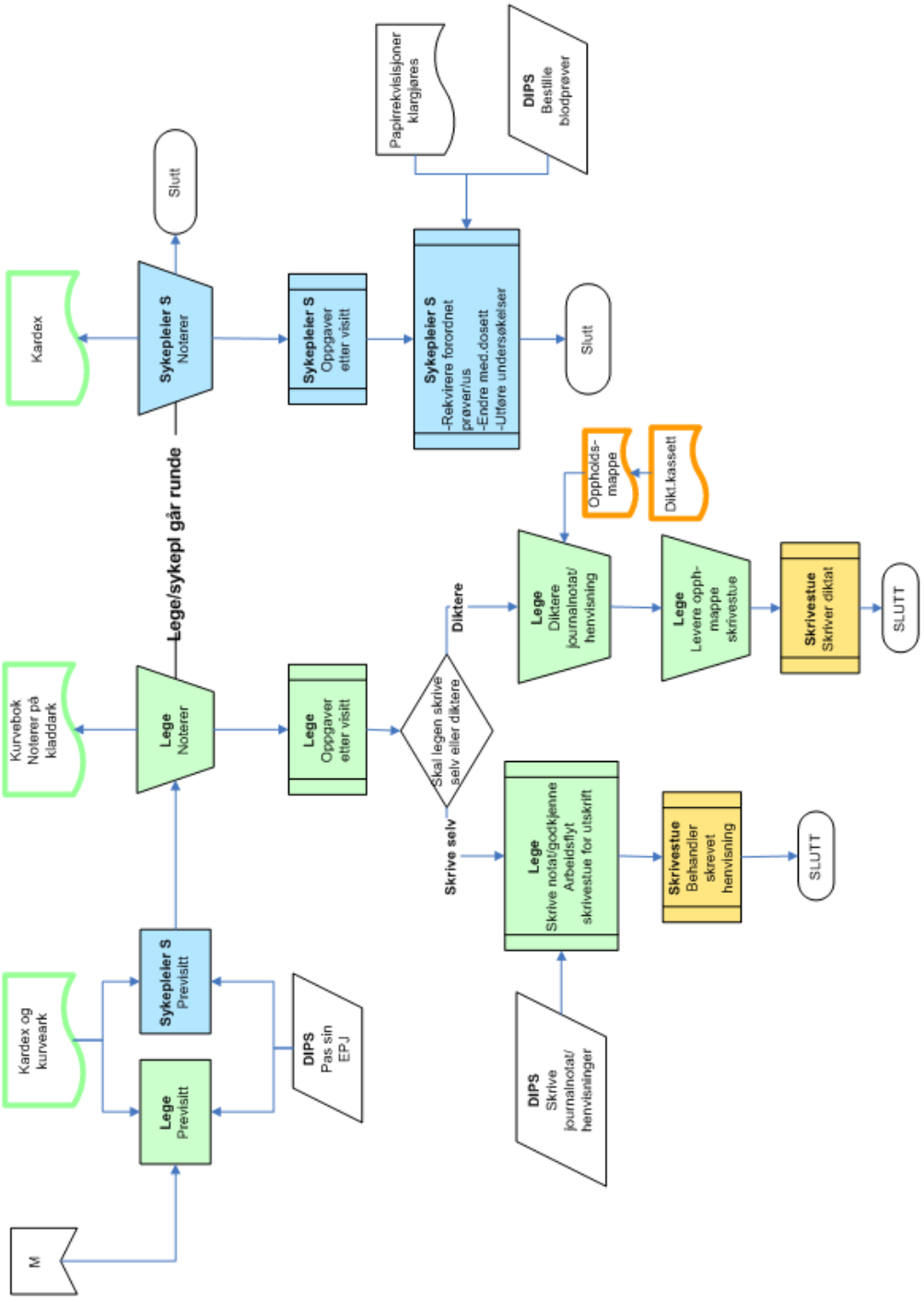

Figure 31: Patient treatment

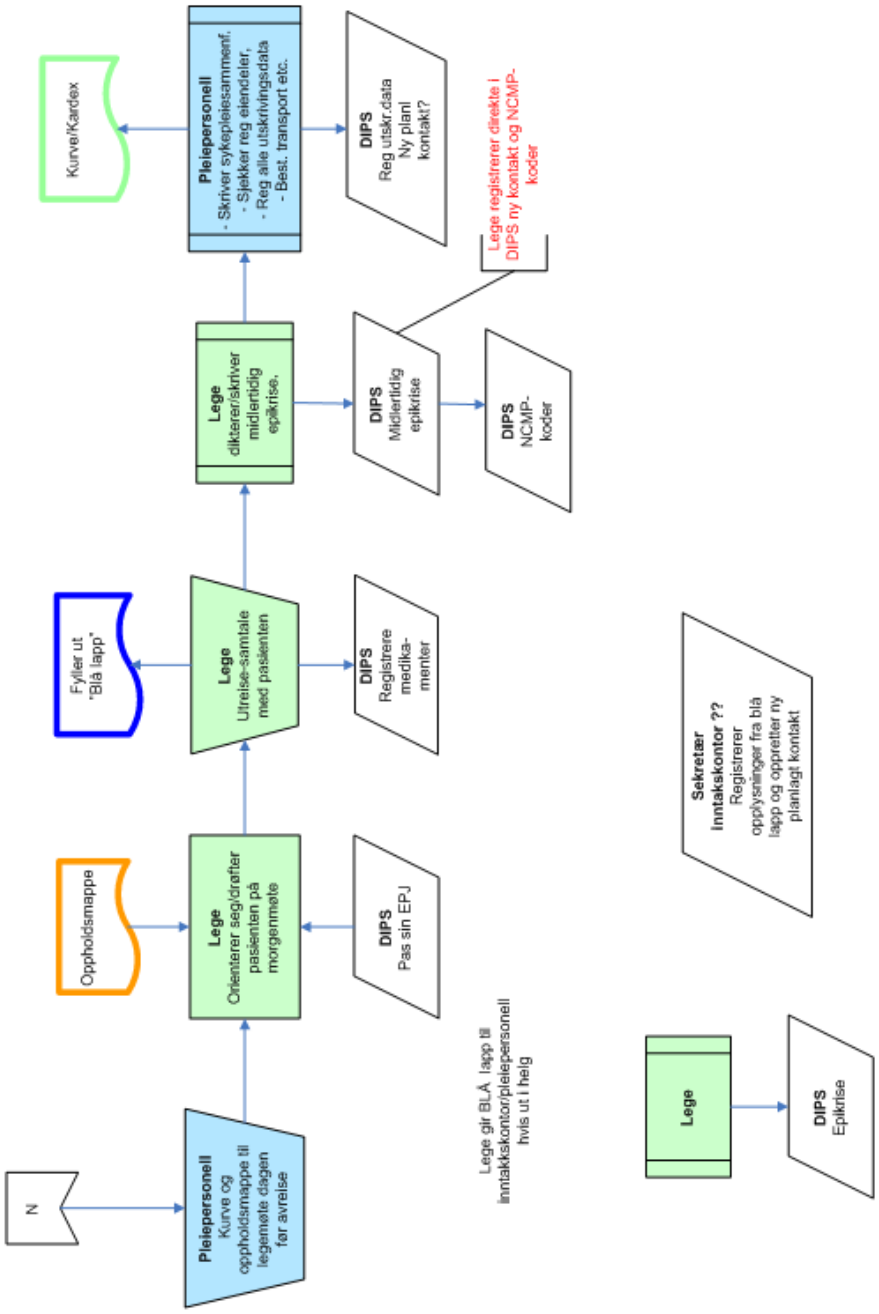

Figure 32: Discharge of patients

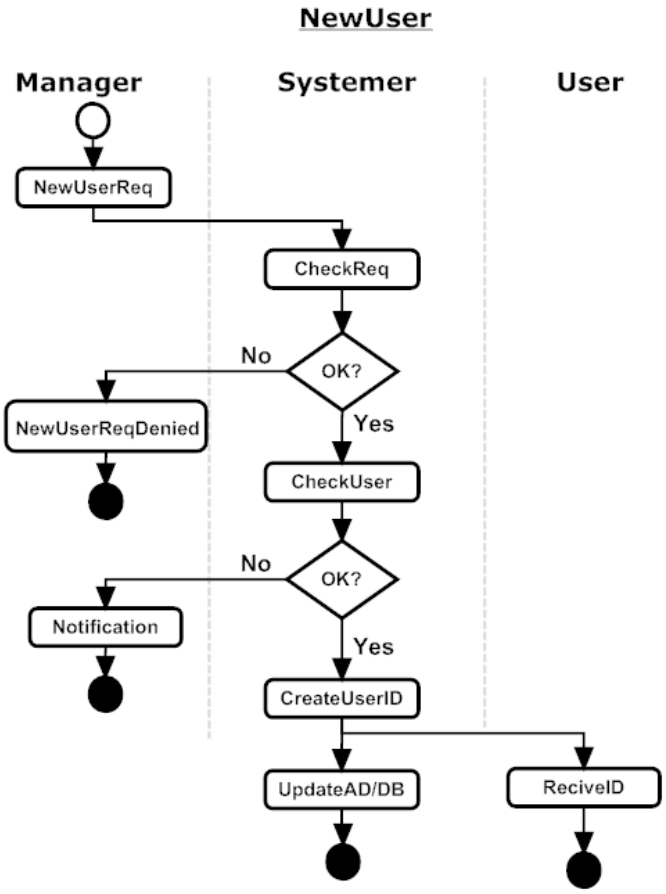

Figure 33: Password - New User

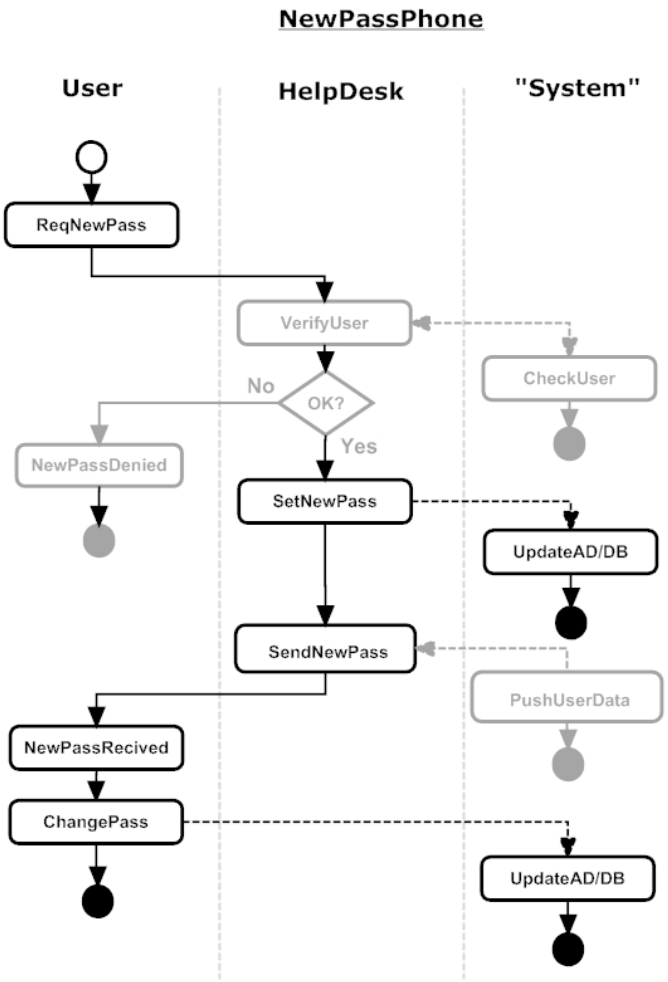

Figure 34: Password - New password over phone

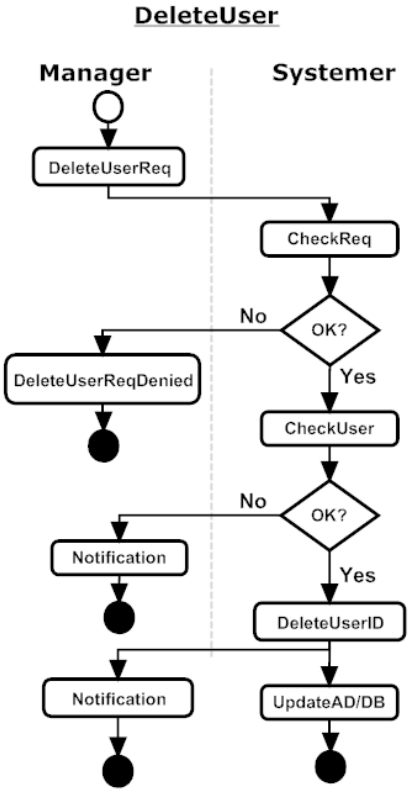

Figure 35: Password - Delete user

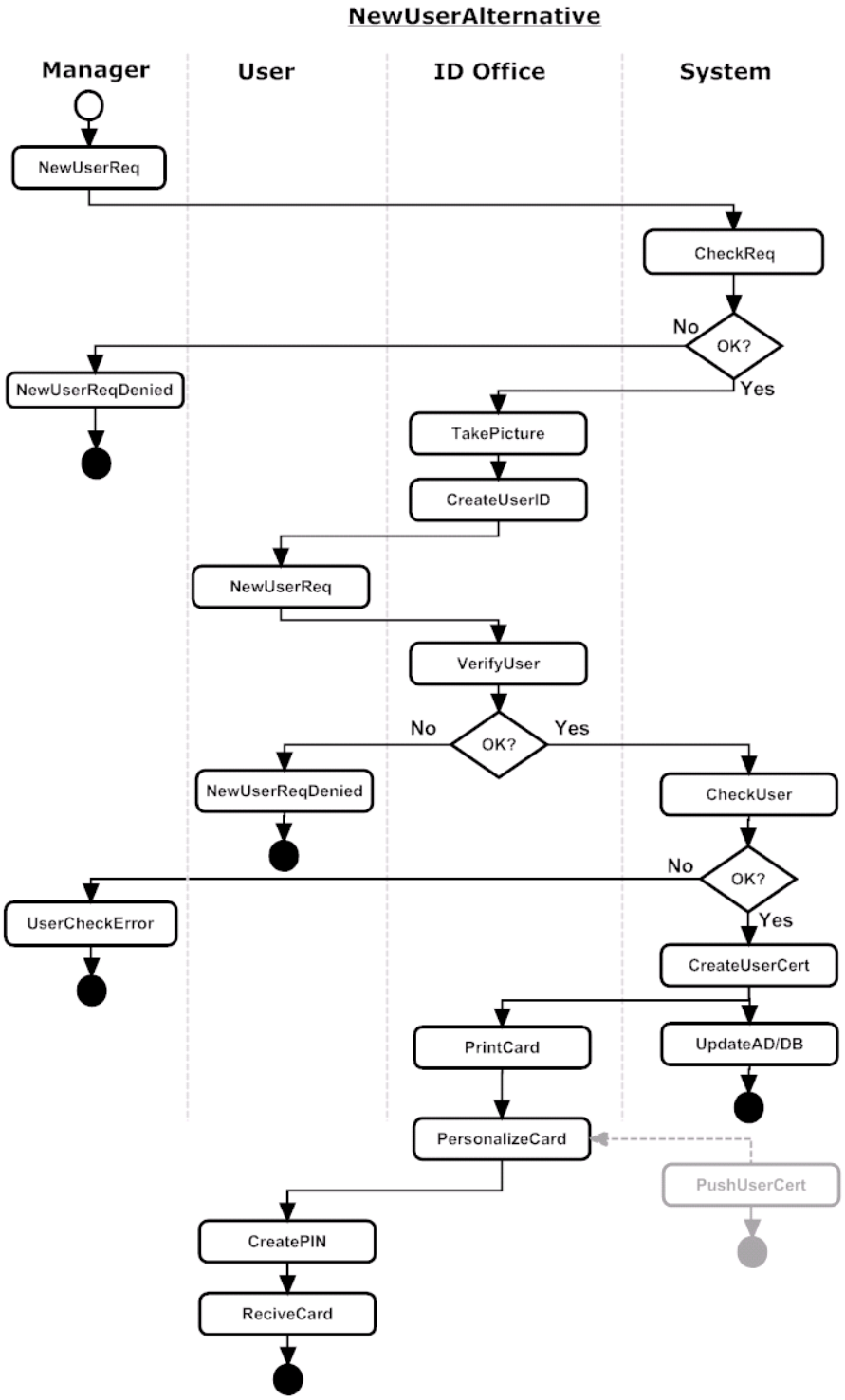

Figure 36: Smart card - New User

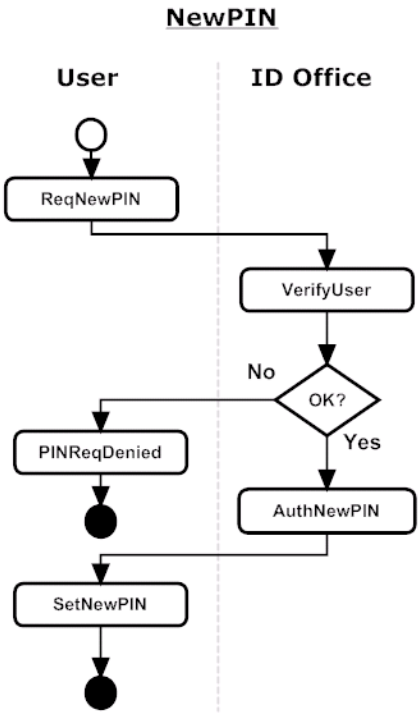

Figure 37: Smart card - New PIN code

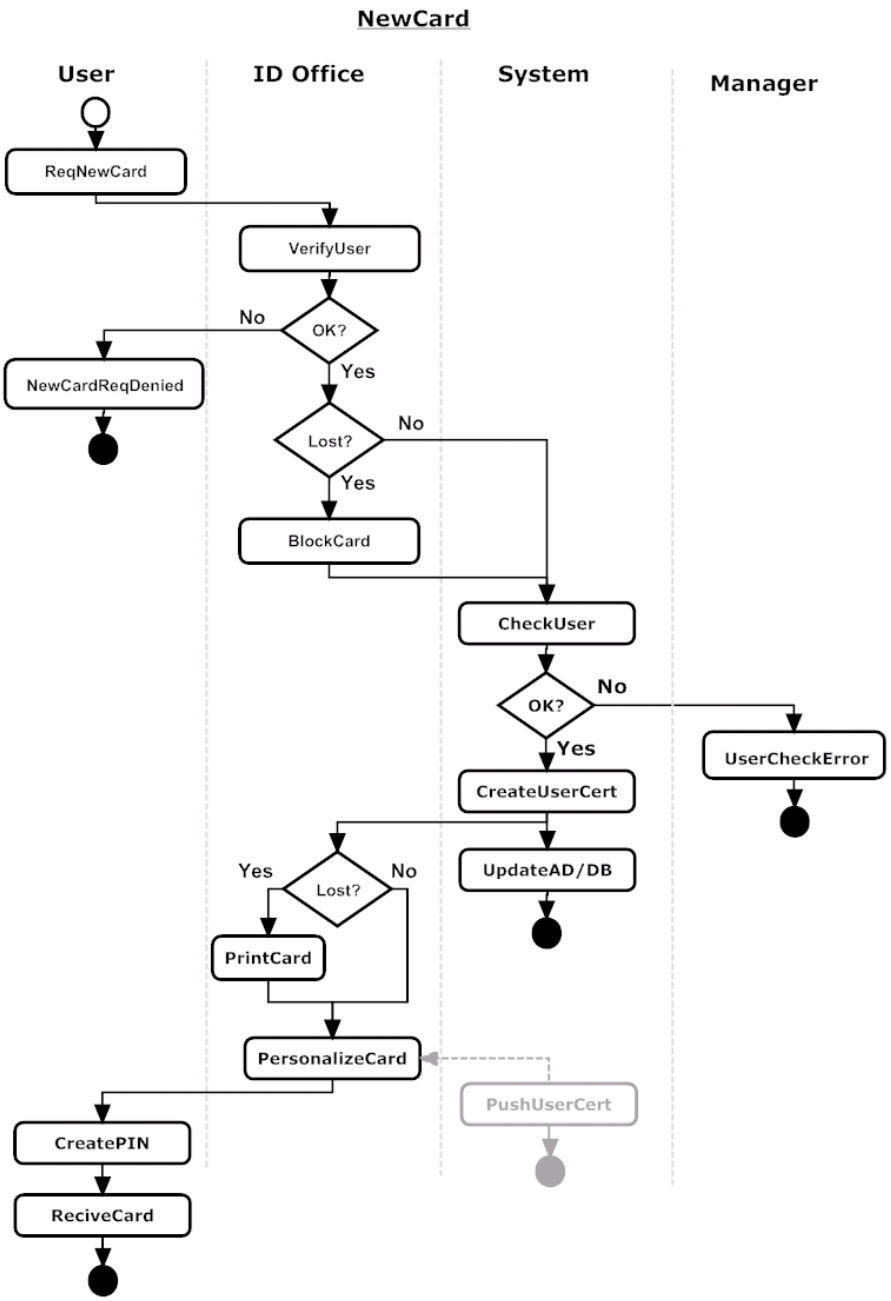

Figure 38: Smart card - New card

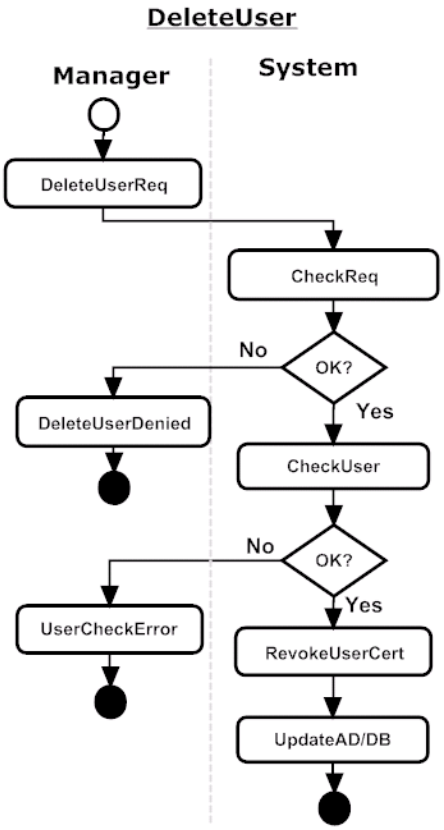

Figure 39: Smart card - Delete user

I Possible States of the Object Doctor

This Appendix describes the state transition performed within the model during a simulation run. The values listed in Table 29 were utilized in the simulation runs when both smart cards and passwords were implemented. The data presented in Table 29 are determined by us and divides the doctors in equal large work shifts.

Table 29: Start values used in model

Fields	**Records**											
PersID	1	2	3	4	5	6	7	8	9	10	11	12
@Home	1	0	1	0	1	0	1	0	1	0	1	0
@Work	0	1	0	1	0	1	0	1	0	1	0	1
PAC ID Office	0	0	0	0	0	0	0	0	0	0	0	0
Available	0	1	0	1	0	1	0	1	0	1	0	1
Know password	1	1	1	1	1	1	1	1	1	1	1	1
Know PIN	1	1	1	1	1	1	1	1	1	1	1	1
Have card	1	1	1	1	1	1	1	1	1	1	1	1
PersType	1	1	1	1	1	1	1	1	1	1	1	1
Work Shift	0	1	0	1	0	1	0	1	0	1	0	1

In order to separate each employee in the model, i.e. doctors, nurses, security guards etc., each object have a unique ID ("PersID"). Since we initialize the model with twelve different doctors, each receives a "PersID". Should we have included the feature of adding new doctors, and removing those that leave, the "PersID" would have incremented based on the previous generated "PersID".

The different state transitions are described below. Furthermore, the different state transactions are also illustrated in Figure 40.

When a doctor is at home, "*@home*" is set to 1, and the "*@work*" parameter is set to 0, and vice versa. As soon as the "*Work Shift*" parameter is set to 1, those doctors that are "*@home*" will go to work, and goes to the physical access control. Should the doctor have forgotten his PIN code or does not have his smart card, he will need to visit the ID office. Otherwise, the doctor becomes available ("*Available*" is set to 1).

Any new business activity (BA) requests can be assigned to a doctor where both "*@work*" and "*Available*" are equal to one. When a request is assigned to the doctor, he no longer becomes "*Available*". As soon as the request is processed, the doctor again becomes "*Available*".

When the work shift is over and as soon as the doctor becomes "*Available*", the status changes from "*@work*" to "*@home*".

Throughout the simulation a doctor could forget his password ("*Know password*" = 0) or PIN ("*Know PIN*" = 0), or lose his card ("*Have card*" = 0), and this status will not change before the doctor has contacted the ID office. The doctor will only notice these situations either as he/she

arrives at work ("*PAC ID Office*") or when a request is assigned to him. In both these situations, he will be "*@work*", but due to the delay, it will take longer time before he is "*Available*" and hence start processing the BA. This delay then affects the processing of requests.

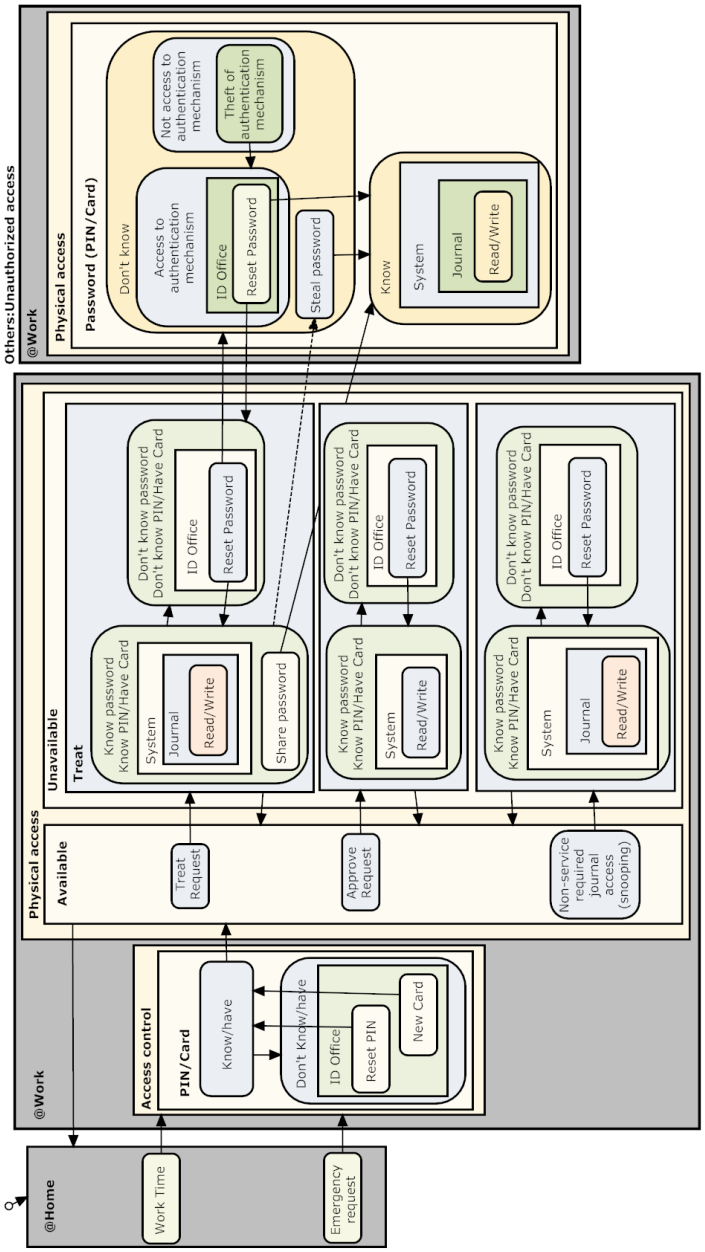

Figure 40: Possible state transitions of the *Doctor* object

J Model Iterations

As mentioned in Chapter 8, the process of developing the MIMD methodology has been an iterative process. Because of this, we briefly present the different iterations of the model in this Appendix.

J.1 The first iteration

Although the overview of the model, depicted in Figure 41, seemingly looks simple, Figure 42 reveals that the model quickly becomes complex. The models structure is directly created based on the flow charts (Appendix H) and each of the tasks identified are included as depicted in Figure 43. The same is true for the processes of the security measure, illustrated in Figure 44 which includes all the steps preformed in order to reset the PIN code of a smart card.

Basically, this first methodology results in a model which becomes too complex which again results in a model that never will be possible to simulate as too many errors will occur. Such an approach will of course not reduce the complexity of the business process to a level which is needed.

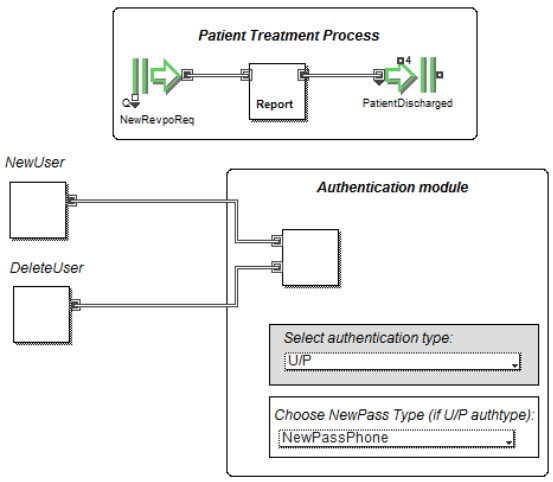

Figure 41: Overview of model - First iteration

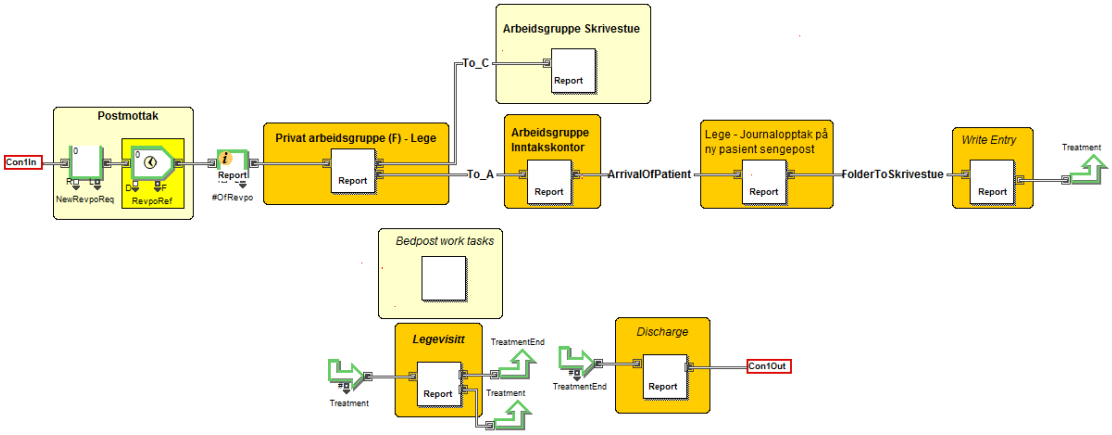

Figure 42: Treatment overview - First iteration

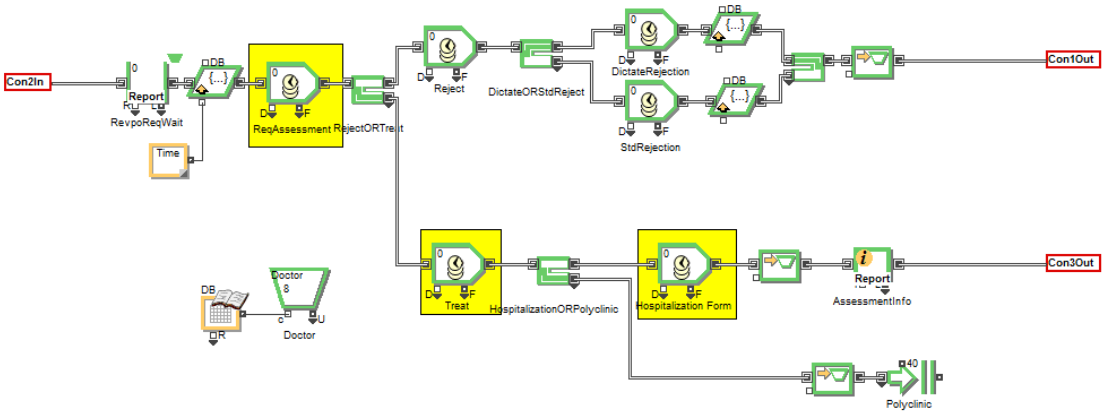

Figure 43: Request assessment - First iteration

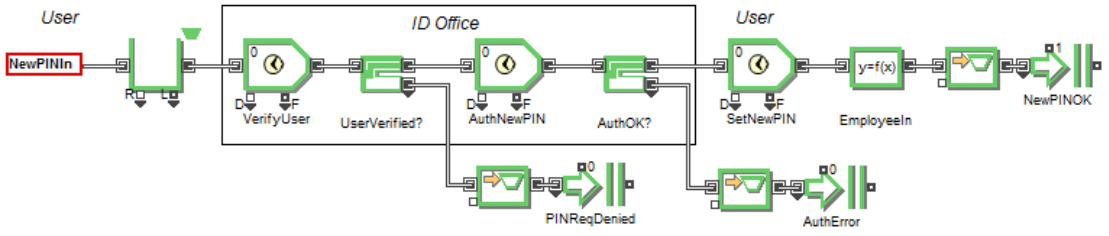

Figure 44: New PIN - First iteration

J.2 The second iteration

Based on feedback from Fredrik Dahl and our own experience, a reduction of the complexity was needed. Based on this, the second iteration focused on eliminating some of the details found in the first iteration. Although the security measure module was kept intact, the patient treatment module was altered with the intention of reducing the complexity.

Comparing Figure 42 and Figure 45, we recognize the connection between the main tasks and that the sequence is still intact, but when we look closer at the same task as in the first iteration (Request Assessment), in Figure 46, we see that the level of detail have been reduced. We are now only concerned about the fact that the doctor performs the task, but nothing more. Basically, all the details have been compressed into one step.

However, the temporal relationship between the business activities was still considered a problem with regards to reducing the complexity sufficiently. The methodology also still presented issues surrounding scalability. Based on these issues, we conducted a third iteration.

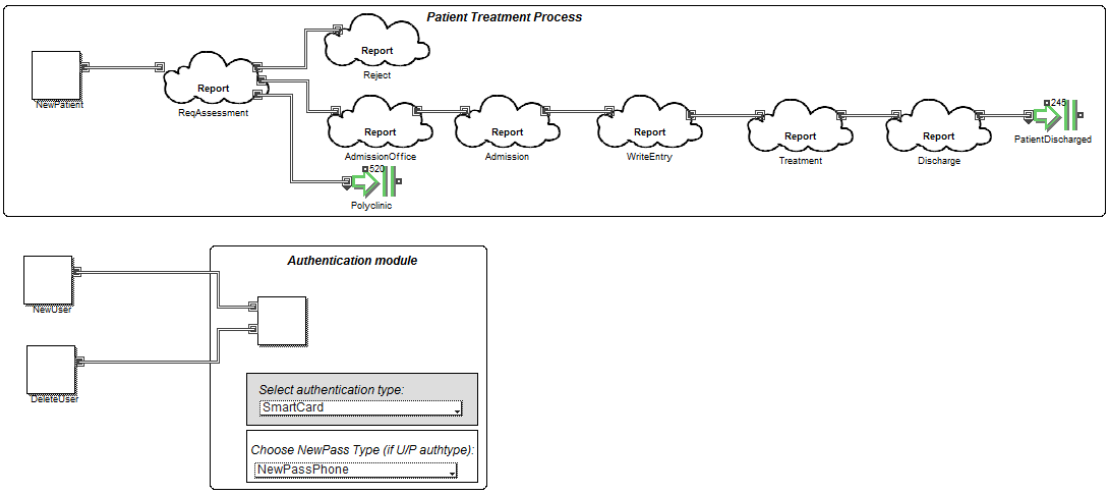

Figure 45: Overview of model - Second iteration

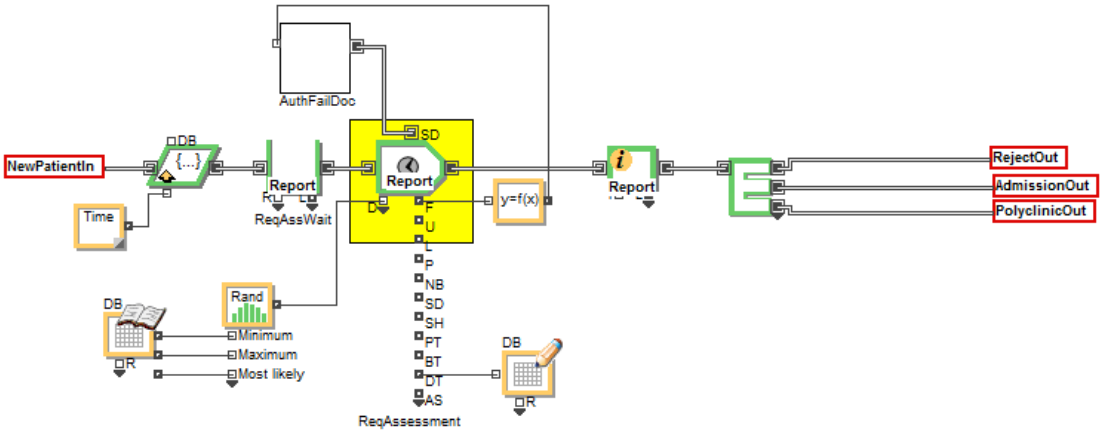

Figure 46: Request assessment - Second iteration

J.3 The third iteration

Several, large, alterations were introduced in the third, and final, iteration of the methodology in order to solve the issues which the former iteration presented. Perhaps the largest change was going from a "flow chart" construction method to a state machine construction method. Another large alteration was to identifying the main (general) tasks and participants, and builds the model around these. A third alteration in order to enable scalability was to identify the common elements and sharing these between the objects/participants. The exclusion of the temporal relationship had by far the largest impact with regards to reducing the complexity of the system to be modeled.

By following the third iteration we first created a model structure as depicted in Figure 47 to Figure 51. However, since this was considered too complex and not possible to scale easily, the utilization of the database was increased. This resulted in the model structure presented in Chapter 7. Although the model functions similar, the main difference between these two approaches is that the state transitions are made directly against the database, and not visually illustrated in the model structure. It is important to notice that the complexity we encountered by utilizing this third iteration of the methodology, was due to design choices made by the author, and not the methodology itself. Because of this, we do not classify the utilization of a database as a fourth iteration.

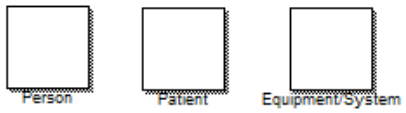

Figure 47: Overview - Third iteration

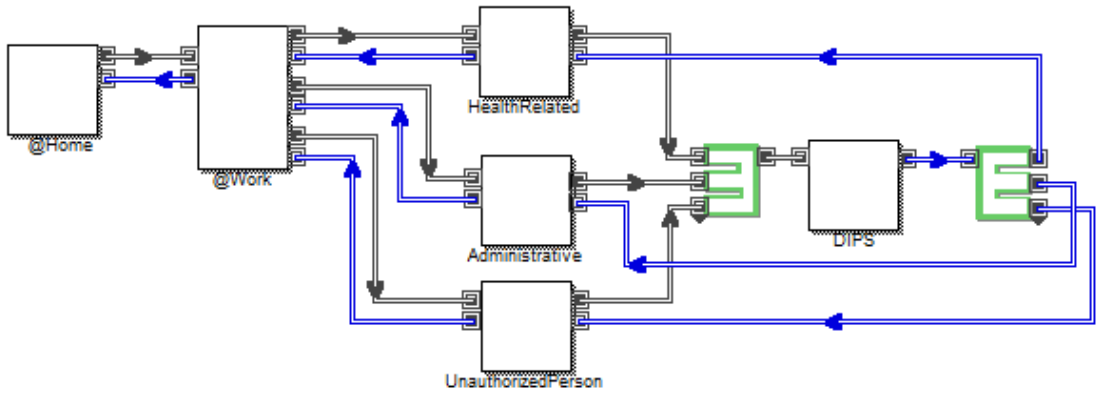

Figure 48: Person - Third iteration

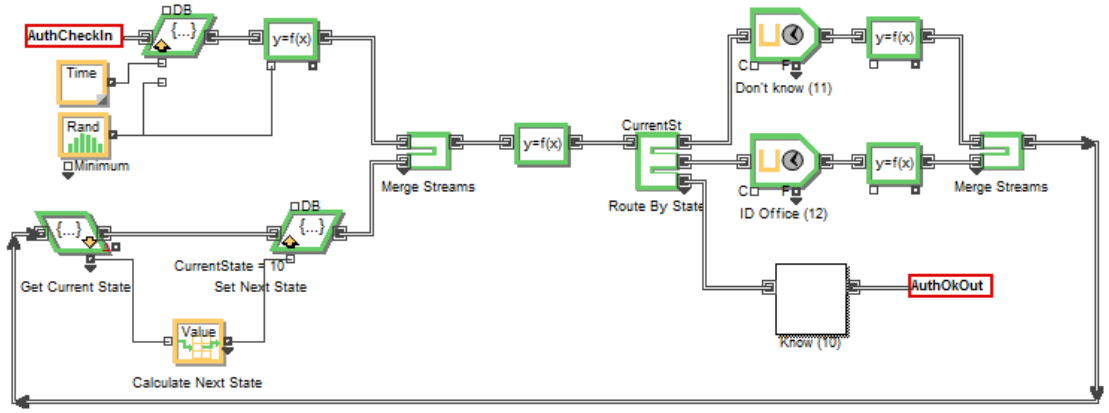

Figure 49: DIPS - Third iteration

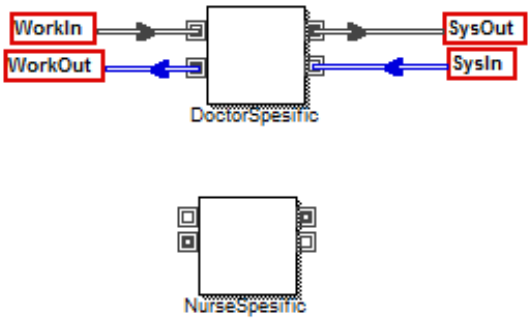

Figure 50: HealthRelated - Third iteration

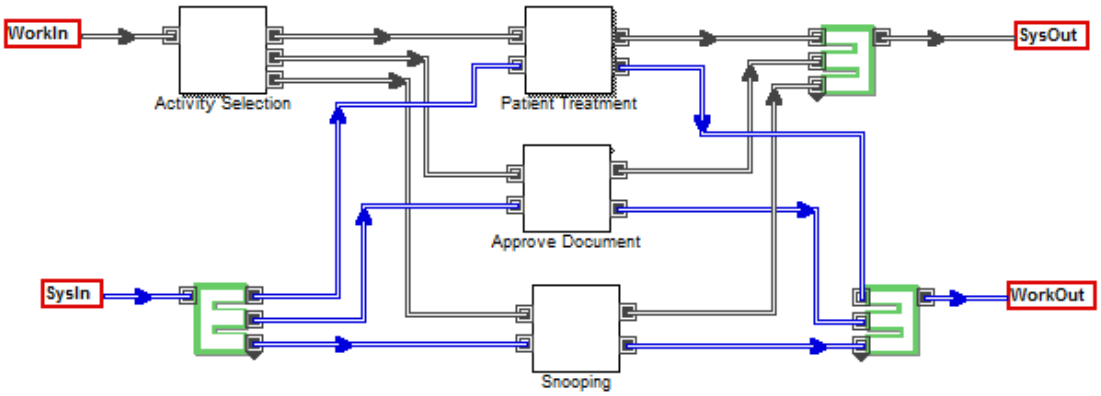

Figure 51: Doctor Specific - Third iteration

K Simulation Output

The simulation output generated during the case study is in this Appendix presented.

K.1 Determining warm up period

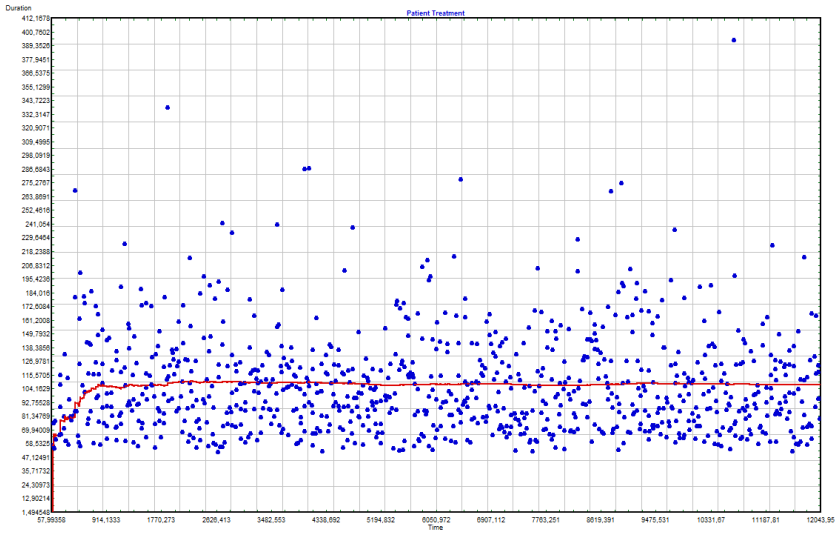

Figure 52: Individual observations and cumulative averages for Patient Treatment with password

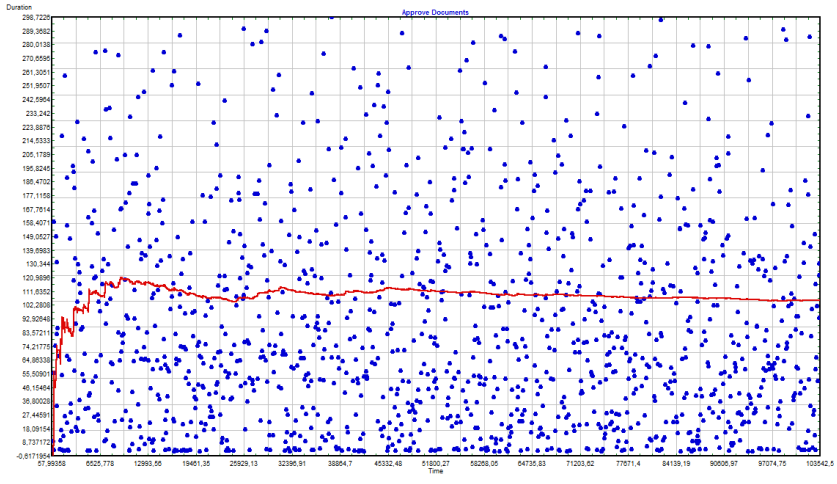

Figure 53: Individual observations and cumulative averages for Approve Documents with password

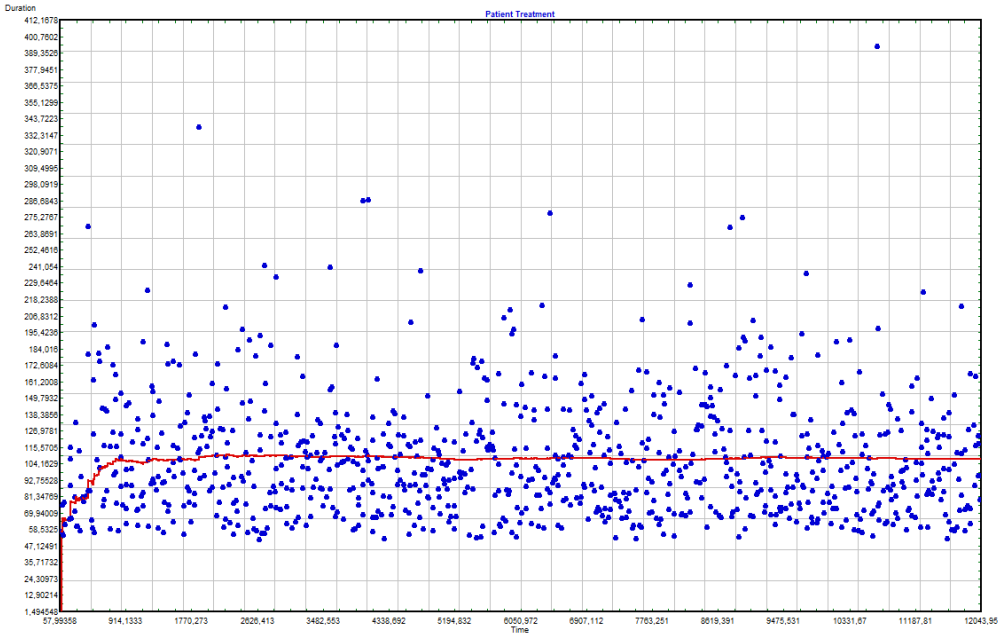

Figure 54: Individual observations and cumulative averages for Patient Treatment with smart card

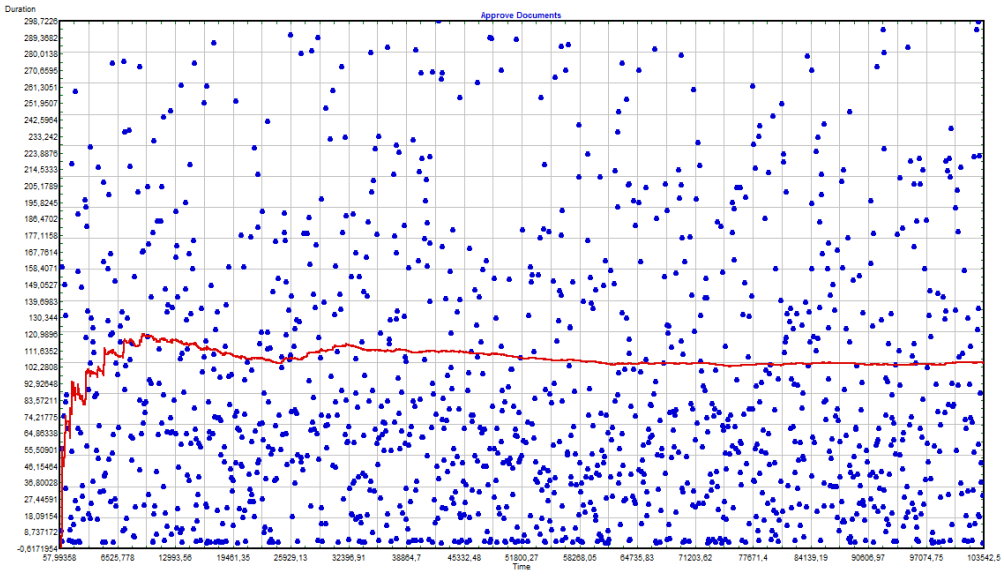

Figure 55: Individual observations and cumulative averages for Approve Documents with smart card

K.2 Analyzing difference between system designs

Table 30: Computing the difference between smart cards and passwords

	Patient Treatment			**Approve Documents**		
Batch	Password	Smart Card	Difference	Password	Smart Card	Difference
1	106,08	106,16	-0,08	97,28	96,27	1,01
2	106,36	106,12	0,24	100,70	94,37	6,33
3	106,28	106,97	-0,69	98,23	99,16	-0,93
4	106,38	105,89	0,49	100,00	97,10	2,91
5	106,99	106,32	0,67	99,25	104,09	-4,84
6	105,20	105,45	-0,24	94,67	97,65	-2,98
7	106,24	105,73	0,51	99,04	98,09	0,95
8	106,39	105,45	0,93	97,06	98,54	-1,48
9	106,64	106,03	0,61	101,35	97,34	4,01
10	106,69	106,36	0,33	93,80	96,04	-2,23
11	107,04	105,94	1,09	93,98	100,25	-6,27
12	105,70	104,86	0,84	101,13	104,00	-2,88
13	106,24	105,81	0,44	97,79	101,03	-3,25
14	106,65	107,70	-1,05	98,91	98,38	0,52
15	106,12	105,96	0,16	97,53	99,44	-1,91
16	105,81	106,23	-0,42	96,72	97,35	-0,63
17	106,37	105,79	0,57	98,44	97,56	0,88
18	106,12	106,83	-0,71	99,92	93,64	6,28
19	106,21	106,64	-0,43	99,23	101,34	-2,11
20	106,86	105,99	0,87	98,26	91,63	6,62
21	106,39	106,84	-0,45	100,64	96,98	3,66
22	106,55	106,37	0,18	98,90	98,88	0,01
23	105,84	107,00	-1,16	99,87	102,48	-2,61
24	105,26	106,35	-1,09	100,24	95,14	5,09
25	105,91	106,68	-0,77	96,53	98,53	-2,00
26	106,15	106,01	0,14	101,20	96,92	4,28
27	106,59	106,48	0,11	95,94	97,16	-1,21
28	106,09	105,48	0,61	92,07	93,94	-1,86
29	106,45	105,76	0,69	102,02	98,46	3,57
30	107,31	107,26	0,05	98,53	98,14	0,39
31	106,10	106,13	-0,03	97,61	99,98	-2,37
32	106,65	107,44	-0,79	96,81	94,98	1,83
33	105,33	106,03	-0,70	101,55	95,14	6,41
34	106,61	106,37	0,25	100,01	100,53	-0,51
35	106,13	106,70	-0,57	100,85	95,37	5,48
36	105,28	106,98	-1,71	101,18	100,63	0,55
37	105,52	107,02	-1,49	96,75	96,69	0,06

L Output Generated from the Ranking Methodology

The output generated from the Ranking methodology when applied on the scenario described in Chapter 6[1] is listed in Table 31, 32, 33, 34 and 35.

Table 31: Non-simulation approach: Security level compatibility

	Password	**Smart Card**	
		Card	PIN
Search space	Characters: 8	0	4
P(sos.eng)	0,5	-	0,5
P(loss)	-	0,02	-
P(gue.pw)	0,002	-	0,03
Entropy	1	5,6	1
Sum entropy	6 (with key)	6,6	
Secure product H > 14	No	No	

Table 32: Non-simulation approach: Needed estimates for usability computations

	Password	**Smart Card**	
		Card	PIN
Working days per year	360	360	
Auth. Sessions per day	11	11	
One transaction (sec)	10	10	
One renew	5 min	15 min	11 min
Num of renews (/user/year)	1,36 (+12[1])	0,37	0,426
Human error (per cent)	60	60	
System error (per cent)	5,56	5,56	

[1] Although we have not included this in our model, one should in addition to the failed authentication attempts, also include the number of times a user is "forced" to chance the password. Assuming that the password policy requires that a user changes the password once a month and further assuming that such a password change takes only 1 minute, 12 minutes are added to see what effect this has on the outcome.

The parameter "num of renews", in addition to the twelve "forced" password changes, also includes those incidents when a user forgets his/her password, PIN or card. The "Human error" parameter is based on [27] and represents the number of times a user needs to retry the authentication process due to misspelling the password or similar. However, these figures does not include those events that results in a user forgetting his/her password/PIN, as these are included in the parameter "num of renews".

[1] Due to the difference between the two methodologies, we will need to utilize additional data which we have not used in our model. When possible, we will use collected but non-utilized data from our data collection phase, however where such data is not available, we will use data found in [27]. We do this as we believe that these figures are applicable in our case and that estimating these figures our self will not provide any better results.

Table 33: Non-simulation approach: Usability of authentication product

	Password	Smart card
Enrollment (min/year)	2	2
Transaction (min/year)	660	660
Renew (min/year)	7 (+12[1])	10
Human delay (min/year)	396	369
System delay (min/year)	36,7	36,7
Threshold: 1,6 min/day	3,06 min (3,07)	3,06 min
Usable products	No	No

[1] Including "forced" chances of password

Table 34: Non-simulation approach: The needed estimates for the cost computations

	Password	Smart card	
		Card	Reader
Infrastructure costs			
Admin. Costs (€/h)	85	85	
Single equip. (€/user)	0	30	7,5
Nr of equip	0	1:1	1,5:1
Software (€/user)	0	0	0
Implementation	0	0	0
Installation	0	-	5 min x NRE[1]
Enrolment (min)	0	-	-
Template Storage	0	-	-
Administrative costs			
Admin. Costs (€/h)	85	85	
Staff turnovers (/year)	2	2	-
Enrolment (min)	5	5	-
Renew	5 min	5 min	-
Nr of renewing	1,36 (+12[2])	0,796 (0,37 + 0,426)	-
Termination	-	-	-
License	-	-	-
Maintenance	-	-	-

[1] Non-Recurring Expense [2] Including "forced" chances of password

Table 35: Non-simulation approach: Cost of the authentication products

	Password	Smart Card
Infrastructure costs (€)	0	262,5
Administrative costs (€)	40,8 (108,8[1])	19,8
The total sum (€)	40,8 (108,8[1])	282,3
The total sum per user	3,4 (9*)	23,5

[1] Including "forced" chances of password