Master's Thesis

Invisible Security

Per Grahn

Department of Electrical and Information Technology,
Faculty of Engineering, LTH, Lund University, August 2015.

Invisible Security

Per Grahn
adi09pgr@student.lu.se

Department of Electrical and Information Technology
Lund University

Advisors: Paul Stankovski & Joakim Ericsson

June 22, 2015

Printed in Sweden
E-huset, Lund, 2015

Abstract

Security is a central aspect in all IT systems, but while security solutions have improved and become more advanced and complex, users' needs and requirements have been put on the back-burner. This master's thesis is an attempt at bringing users back into focus when discussing computer security, and to propose a security platform that is less distracting to users' work.

The thesis has taken a user-centered approach to its design and a survey of user behaviour and needs while working against an IT system was conducted with contextual interviews on users in three different areas; retail, health-care and corporate. The contextual interviews resulted in sequence models for interesting user behaviour and their needs were analysed and composed into a few key requirements for an invisible security platform.

Based on the user survey, a design proposal was created using concepts such as proactive authentication, SSO, session roaming and operation guiding. The design also included a few non-functional requirements which need to be satisfied in order to keep users from being distracted by the system.

Keywords:

Access Control, Computer Security, Contextual Interviews, Human Automation Cooperation, Human Computer Interaction. Invisible Security, User-Centered Design.

Acknowledgements

I would like to thank everyone who has questioned me while writing my thesis; The people at Sogeti, especially my supervisor: Jesper Kråkhede, who never ceased to provide input and ideas. Furthermore I would like to thank my fellow students; Jesper Funk and my reviewers for all the feedback and discussions. Lastly I would like to thank my supervisors at LTH: Paul Stankovski and Joakim Eriksson for holding me back, and making sure I actually finished.

To all my family and friends, thank you for being there.

Per Grahn — 2015

Table of Contents

List of Figures

Abbreviations and Symbols

AAA Authentication, Authorisation and Accounting

AIC Automation induced complacency

ASP Authentication Service Provider

ATM Automated Teller Machine

BYOD Bring your own Device

DAC Discretionary Access Control

HAC Human-Automation Cooperation

HCI Human-Computer Interaction

IAAS Infrastructure as a Service

IBAC Identity Based Access Control

MAC Mandatory Access Control

NFC Near Field Communication

NFR Non-functional Requirement

QUPER Quality Performance

RBAC Role Based Access Control

SAAS Software as a Service

SP Service Provider

SSO Single Sign-on

UCD User-Centered Design

Chapter 1
Introduction

Security is a central aspect in all IT systems. As IT solutions become more advanced, demands on users to make informed and responsible decisions with regard to security rises. Users often lack the training required to make these decision and they cost both time and energy, which can lead to frustration and reduced productivity. As [1] demonstrates with activation during shopping in 3D Secure solutions, when focusing on other tasks, people are not in the right mindset to make good decisions regarding security.

Cyber-espionage has been constantly trending upwards since 2009, putting more pressure on companies security solutions. What is worse: according to [2] at least 10% of all data breaches in companies 2013 were due to employees misuse or other unintentional errors. About two-thirds of these breaches were not discovered by the breached company. Furthermore, after years of touting on how and why to select strong passwords, in 2012 weak or compromised passwords accounted for 76% of all network intrusions[3], and in 2014 the 25 most common passwords still accounted for 2.2% of all passwords on the Internet[4].

If companies keep moving the security solutions on to their employees, the systems will become overly complicated, and although security training employees is important, it can become too taxing and there is a risk they will instead look for less secure shortcuts just to get the job done. A security platform should be unobtrusive and assist employees to keep their work secure instead of keeping them from working.

1.1 Goal

The overall purpose of this thesis' study is to design a concept of a *responsive* enterprise security platform. The security in the platform should be *invisible*. In this case responsive means that the concept should be generalizable to support bring your own device (BYOD) and distributed work-environments, as well as different kinds of authentication. Invisible security is defined as non-distracting security where users are not expected to confront security decisions that require special training.

1.1.1 Research Questions

The analysis presented in later chapters will discuss the overall goal as well as address these research questions:

- How do you design a platform which handles security without distracting users from their main work-tasks?
- Is it possible to retain security while lowering user interaction?
- Which decisions can be automated, to what extent and in what situations?
- How are users' trust and mental model affected when user interaction is reduced?

When doing this, an evaluation of which security decision are important for users with low knowledge in IT-security is needed. So is an investigation in how users are affected when automating security.

1.2 Related Work

A vision of invisible security, similar to the one presented in this thesis has been published by Intel Labs in [5]. The focus in their paper is on the home user. As such some of the privacy considerations are not as applicable in an enterprise environment. They base their paper on the notion that the authentication of today is based on the technology of yesterday and they share their vision on how invisible security could work in everyday scenarios. They also include the main barriers they have identified that have to be solved before their vision could be made reality. They are separated in three areas and consist of the following:

Technology

- Low powered, small biometric sensors.
- Algorithms for combining multiple sensors.
- System integration, cross-device support.

Business

- Hesitation in opening up identification and authentication between companies.
- Concerns in competitiveness of devices and services.
- Cross-company integration is expensive. But ecosystem boundaries must be overcome both from a business and a user perspective.

User Experience

- Privacy and integrity concern over world wide market.
- User trust and system observability.

1.3 Delimitations

Due to the wide scope of this area, a number of delimitations have been put on the thesis study and analysis.

The focus of the study will be on access control: When users are arriving at their workstation, when users are performing operations and, to a lesser extent, when users are leaving their workstation. In the spirit of user-centered design, this thesis will be centered around the users perspective. The security aspects, unless otherwise stated, are assumed to be up to the standards of today.

1.4 Outline

This report is divided into 6 major chapters and a number of appendices. The remainder of the report is structured as follows:

Chapter 2
Describes the underlying theory used as a basis for this thesis. It has been divided into three parts; HCI and user centered design, automation and security related theory.

Chapter 3
Contains results and analysis from the identification of end-users' needs and current style of working, which was carried out with contextual interviews. The results are presented as sequence models.

Chapter 4
Presents the design decisions made for the concept of the security platform. Results are presented in flowcharts and non-functional requirement-diagrams based on the QuPer model.

Chapter 5
Discusses and analyses the results in this thesis and the recommendations of future work to be done on the subject.

Chapter 6
This chapter is a summary of key elements from previous chapters and presents a conclusion of this thesis paper.

Appendix A
Consists of the personas used to help discuss and analyse the result of the user-survey presented in chapter 3.

Appendix B
Contains a legend for flowcharts. It is included to explain terminology and appearance of the flowcharts in chapter 4.

Chapter 2
Theory

"In short, all predictions agree
that if man does not master technology,
but allows it to master him,
he will be destroyed by technology."

— ICRC (1987) [6]

2.1 Human Computer Interaction

Human Computer Interaction (HCI) can be defined as:

> "[T]he design, evaluation, and implementation of interactive computing systems for human use and ... the study of major phenomena surrounding them."[7]

According to Norman in [8], when a user interacts with a computer, the user creates a mental model over how the computer works internally. Interface design can support HCI with interaction that conforms to users defined mental models or supplements mental models that support intuitive interaction. It can also confuse if the user's mental model and user interaction suddenly doesn't match. A well made representation aids us in our mental models, as opposed to the interaction that thwarts our thought process by impelling us into mental states unsuitable for the situation. Norman explains in [9] that it is the *things that makes us smart*. Using computers (and other artifacts) greatly enhances the mental capacity of users.

To aid in designing interfaces for HCI a number of best practices have been established. Among the most recognised are Shneiderman's eight golden rules for interface design, described in [10]. These practices are:

Strive for consistency
: Keeping a consistent terminology, layout and feedback across the system creates familiarity. As does keeping order of sequence in similar situations. Exceptions to the ordinary flow should be comprehensible and kept as minimal as possible.

Cater to universal usability
Users have different needs. Experts might want shortcuts that allow for faster navigation or pacing, novices need additional support to get into the system. recognise the user diversities and aim to support everyone.

Offer informative feedback
Every user action should give feedback. The extent of feedback should depend on frequency and impact of the action in question.

Design dialogs to yield closure
User interaction should be grouped into beginning, middle and end. Giving informational feedback at the end gives user a sense of relief and allows them to drop it from focus and mind.

Prevent errors
As far as possible, prevent users from making decisions which result in errors. This can be done by disallowing different kinds of input where it's not applicable. If an error occurs allow users a clear path of recovery.

Permit easy reversal of actions
Allowing users to undo actions and decisions relieves anxiety and encourages exploration of the system which in the long term leads to expertise of the system.

Support internal locus of control
Users should control the interface. The interface's behaviour should not change, data-entry should be simple and it shouldn't be difficult to obtain information from the interface.

Reduce short-term memory load
Avoid situations where users need to remember information between different states in the system. Provide a clear navigation of where users are in the system and how they got there.

2.1.1 User-Centered Design

User-Centered Design (UCD) is a method for designing HCI-solutions. In [8], Norman describes the main principle of UCD as keeping the users involved throughout the design process. When users are not available personas can be used in their stead like described by Cooper in [11]. The processes involved in UCD has been standardised by ISO in [12] to include the following six principles:

- The design is based on an understanding of users, tasks and environment
- Users are involved in the design and development
- The design is driven and refined by a user-centered evaluation
- The process is iterative
- The design addresses the whole user experience
- The design team includes multidisciplinary skills and perspectives

Critique

UCD is not without critique. Norman raises the point in [13] that over-reliance on user-input can lead to overly complex systems with features many users may not actually use or need (but when asked, want to have). Furthermore, Dearden et al. points out in [14] that designing HCI with a specific end-user in mind, although providing benefits for the targeted group, can lead to disadvantages for others and, according to Abras et al. in [15], by making design user-centered the designers lose overall perspective of workflow, thus excluding indirect stakeholders. Complying different products with different stakeholders of different cultures creates higher isolation of the products, which can lead to more competing standardisations. Abras et al. also points out that these factors contribute to the overall higher time and effort which raises cost required for UCD when compared to e.g. feature driven design.

Norman raises another argument against UCD in [13], which is the ability of humans to learn. If a system can do an activity really well; the struggle to learn this system might be worth the effort. UCD aims to minimise the learning-curve and workload of users, sometimes at the cost of system performance in the hands of an experienced user. Exploiting human adaptability is not inherently bad.

2.1.2 Contextual Design

A variation of UCD is contextual design. Contextual design has its foundation in Holtzblatt's idea, which she describes in [16], that users cannot completely tell what is good and bad in a system. Gathering user design-data without losing details needs to be done while users are actually working on the system. By observing and interacting with users in contextual interviews as they are working in their normal work environment designers are able to see patterns and details that would otherwise be missed. The purpose of these contextual interviews is to get samples of concrete data on how users are actually working as opposed to what company policies and other norms tell them they should (and users would relay on to the interviewer). By observing users' work as it unfolds it is also easier to identify indirect stakeholders.

Sequence Model

A user's actions in a system are not random. They follow a specific pattern as they unfold over time, and the sequence in which actions take place reveals their strategy in solving a task. By following this sequence, the user's intents can be uncovered. Intents may not be obvious at first glance, therefore revealing them in a sequence model can provide valuable information for system designers. To do this Holtzblatt proposes drawing Sequence models, like the one in figure 2.1.

The sequence model differs from similar models like flow diagrams and task analysis by stating intents and triggers for each sequence explicitly. A breakdown or detour from the original intent is marked with a lightning bolt.

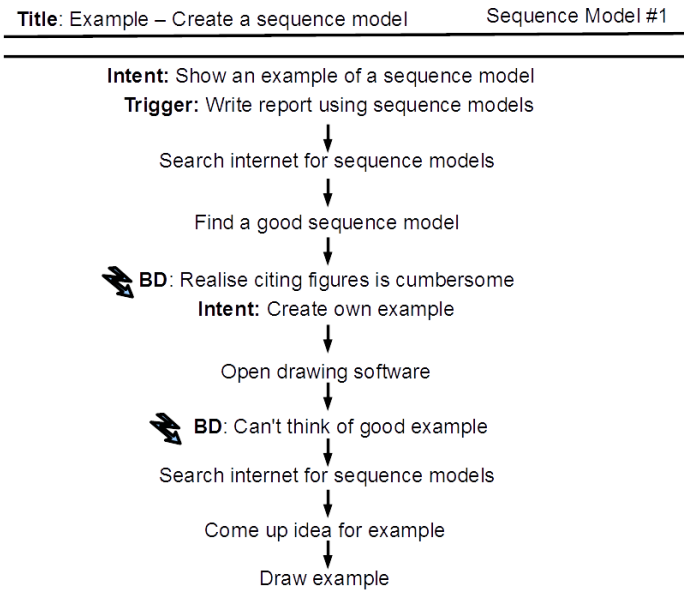


Figure 2.1: Example of a sequence model

2.1.3 Persona

Personas were first proposed by Cooper in [11] as a way of creating a baseline for discussion in a design group. According to Adlin in [17], there are two kinds of personas: Ad-hoc and data driven. Ad-hoc personas are less formal and designed around their creators' expectations and experiences around users. Advantages of Ad-hoc personas are that they are quick and effective to create and help externalise preexisting presumptions the design team has about their target users. Disadvantages are that they might not actually be representative of the target users at all. Data driven personas allows more accurate representations of the target users, at the expense of cost in creation.

A common mistake when writing personas is making too many. Holtzblatt recommends in [18] that there should only be one persona for each key-role in a project. When there are more than one persona for each role or when different members of a design group want very different things in their persona there is a high risk that the focus of the project is not well-defined enough or that the design members has radically different views on the goal. This is usually a cue to go back and reanalyse the data, purpose and goals for the project.

2.1.4 Distraction

As described in the sequence model section, all tasks are performed with a goal in mind. HCI (when designed properly) can become a spontaneous tool in reaching the goal. But, as Norman points out in [9], when a cognitive artifact calls attention to itself, it creates a breakdown in the workflow. For example, When retelling a

story we will often not only gesticulate, we will also use artifacts as representations for what we are describing. A cellphone could be an excellent representation of a shoe(!). But when the cellphone suddenly vibrates the illusion is broken.

As explained by Bødker in the Scandinavian model of activity-theoretical HCI [19]; computers are mediators in our daily lives. Interaction with a computer is not done with the purpose of using the computer, it is used in order to achieve a goal. The subject of the goal is normally not the computer either. Our goal to communicate something with a co-worker will be achieved by writing an email, reading news and online shopping are also merely mediated by the computer. Once we left the learning phase, little conscious effort is spent on our interaction with the interface itself. Reeves and Nass explains this in [20] as the media equation; we have a spontaneous and natural response to media in the same way as social interaction with other people.

Every time we shift our attention from our goal we do a focus shift. A focus shift can occur either from external distractions or from internal distractions. An internal distraction can be a constraint from the artifact or unexpected behaviour from it (bug or user misuse). Focus shifts can occur frequently and Reeves and Nass has shown that their impact depends on how related tasks are to each other, how long a user work on a task before the focus shift and how long the interruption was.

2.2 Automation

Automation has been and continues to be a highly controversial subject. Parasuraman and Riley defines it in [21] as "the execution by a machine agent (usually a computer) of a function that was previously carried out by a human." While some, like Christoffersen in [22], highlights the benefits of automation; increased productivity, efficiency and minimising human error. Others, like [23], argues that yielding (too much) control to computers leads to knowledge decay. Norman warns in [24] that it can lead to loss of situational awareness among operators and, if you believe [6], automation will ultimately lead to the demise of human civilisation (as we know it).

Welcome to the Matrix have a nice stay!

However, it is generally agreed upon [e.g. in 21, 24, 25] that automating a task does not merely supplant human interaction but fundamentally changes it. Parasuraman and Reeves points out in [21] that automating a task can lead to better mental workload, but also worse if e.g. presentation is not aligned with users' mental model. U.S Federal Aviation Agency describes in their design standard [26] that automation should only be done to improve system performance, without reducing human involvement. They also recommend training users in when they should question automated systems.

As described in the distraction subchapter of this thesis, a breakdown in workflow occurs when an artifact brings attention to itself. Automation works in the same way, a change made by an automated system creates a breakdown in its operators workflow, and if serious, as described in [10], it can cause the operator to lose orientation with risk of severe consequences.

2.2.1 Automation Induced Complacency

In some cases operators of highly automated systems experience complacency. 'If this task is automated, why should I care about it?'[27] Bailey et al. shows in [28] that automation induced complacency (AIC) increases with system reliance and it has a direct relation to trust and loss of situational awareness. High trust in a system also has high AIC which leads to lower grade of attention spent observing the system, especially if they are occupied with other tasks. When automating a task for an expert, Culley points out in [29] that the trust will depend on the expert's trust in the designer of the automated system. 'Do I trust the designer to understand the intricacies of this task?'

According to Culley, one way to lessen AIC is to variate the degree of automation in a system. The system will then sometimes require user input and sometimes it will do it by itself. But this can lead to a *'cry wolf'* effect. i.e. user latency for when the system fails becomes longer, since user input is no longer only needed for critical tasks. Instead it is preferred to calibrate the resolution of automation to user expectations.

Lee et al. describes resolution in [30] as how the capabilities of the system maps to users trust in it. With high resolution users trust in components are fairly separated and trust in certain components are not affected by performance in others. In low resolution users trust in the system is seen as one entity and therefore performance of components affect trust in others. If resolution is poorly calibrated, users trust in a system does not match with the system's capabilities. This leads to, as described by Parasuraman and Riley in [21] and shown by Dzindolet in [31], either misuse or disuse of the system.

Norman describes how good calibration can be done by providing relevant feedback in [24], and Christoffersen adds the importance of providing high system observability in [22]. Human-automation cooperation (HAC) should provide similar feedback as human-human cooperation. As described by Reeves and Nass in [20], we know that automation is not human, and we do not expect it to be, but we still act towards it as if it is! Some might take this to mean that automation agents should be designed to resemble humans, which is not at all the case. But as Christoffersen [22] and Lee et al. [30] describes, HAC should provide the same cues in information transferal as human communications do.

2.2.2 Monitoring

Parasuraman has also shown in [21] that performance of monitoring automation while simultaneously performing other tasks is usually poor. Performance is also consistently poor both among operators who are accustomed to automated systems and those who are not. There is also a big problem with alerting users of monitoring due mistrust caused by false alarms. Lee highlights in [32] that this is not a trivial problem to solve, since false alarms are beneficial in keeping operator preparation up to standards. If an alarm would alert at 100% hitrate, operators might not recognise the alarm in time and be unable to figure out the correct sequence of actions to correct this problem.

Experiments performed by Jamson et al. in [33] with smart cars showed

that operators are reluctant to override automated behaviour even when they acknowledge behaviour where they could improve performance (e.g. overtaking a slower car in front of them). This could be either because of AIC but could also be explained by the media equation; it is impolite to interrupt the automation. Furthermore they showed that monitoring performance were dependent on whether conditions were considered easy or difficult. High traffic areas were closer monitored compared to light traffic areas. Lastly they showed that monitoring during less demanding conditions lead to higher fatigue of operators.

Observations made by Kircher et al. in [34] highlights that monitoring automation is not a reactive activity but an interactive one. Operators include automation behaviour in their tactical planning of the situation as a whole. This goes in line with the notion that HAC fundamentally changes a task and emphasises that HAC should provide information in a way suitable to the operators.

2.2.3 Levels of Automation

There are multiple distinctions and separations between levels of automation. One example is provided by Parasuraman et al. in [27]. Many proponents, like [23], for these distinctions want to use these levels as a mean to avoid over-automation of tasks. But, Bradshaw et al. argues in [25] that models based on these levels of automation are not only wrong and counterproductive, but also dangerous towards design and discussion around automation as a whole. Furthermore they claim in [35] that trying to define different kinds of automation as more or less on a linear scale does not help designers in a meaningful way instead it paints them in a corner. It is not helpful, and sometimes even impossible, to say that this automation is *more* automated than that automation. Therefore Bradshaw et al. recommends that models with levels of automation should be kept as far away as possible from designers.

2.3 Computer Security

Verizon's Data breach and investigations report [2] have identified users as involved in some capacity in just about every breach recorded. It can be failure to apply a patch, users' mistakes, failure to comply with policies or intentional malicious behaviour. A system is only ever as secure as the users that use (or, according to some programmers; abuse) it. However, users can also be the greatest asset. More breaches have be discovered by a company's own users than any other technology or process. Therefore, it is important to know your users, what they are allowed to do in the system and what they actually do.

2.3.1 Access Control

Karp defines access control in [36] as "Access Control is the mechanism in which the services know whether to honor or deny requests." Traditionally access control has been isolated within a domain. E.g. a single application, website or company, but nowadays this is not necessarily true.

Access control is usually differed between discretionary access control (DAC) and mandatory access control (MAC). As the orange book [37] describes, in DAC the owner of an object sets permissions and restrictions while in MAC these permissions and restrictions are decided by the system. These two versions can be used together.

A basic model of access control is identity based access control (IBAC). IBAC authenticates users (henceforth known as subjects) on an individual level and applies authorisation based on their individual privileges. This is quite cumbersome to scale up as every new subject need to define their privileges for every differing operation and object in the system.

Role based access control (RBAC) is a NIST-standardised [38] model for Access Control. Compared to IBAC, RBAC assigns individuals roles, and applies authorisation based on a predefined set of privileges carried by their role. It is commonly used in enterprises today as it allows relatively simple means of access control with better scalability as well as easier maintainability compared to IBAC. Although RBAC still has problems with scaling across domains.

AAA – Authentication, Authorisation and Accounting

Authentication, authorisation and accounting (AAA) is a commonly used framework for handling access and enforcing policies in computer networks.

Authentication is the identification of subject and/or host. Authentication is commonly done with username/password combination, but it can also be done by providing a token, biometric data or recognising user-pattern. In [39], Gollmann describes these as providing something we know, something we have, something we are and something we do respectively. If we provide more than one of these in a multi-factor authentication we can be more sure that we authenticate the correct subject. Meanwhile, mutual authentication is achieved when both subject and host authenticate each other.

Authorisation determines what rights a subject has in the network, which actions are permitted and what resources are available to them. According to [40], Authorisation can be viewed as the result when evaluating a subject against the networks policies.

Accounting is the logging of user activity. It is needed to hold subjects accountable for their actions and making sure policies are followed and contracts are kept. Accounting is important for maintenance purposes, auditing of the network and performance evaluation. Accounting is also the key component in data forensics. Because no matter how prepared a company is, incidents will happen.

2.3.2 Single Sign-on

Single Sign-on (SSO) services allow a subject to authenticate only once no matter how many services they are registered to. Gollmann warns in [39] about the

curse of convience in SSO, i.e. making a system more convenient to use opens up new angles for threats and attacks. Therefore SSO can't just be added to traditional access control. Pashalidis and Mitchell describes different solutions to SSO in [41]. In most of these solutions an authentication service provider (ASP) is needed. There also need to be an established trust between the ASP and the different service providers (SP). When a subject authenticates to a SP they claim that the ASP can vouch for their identity and the SSO scheme used provides means for a SP to confirm this claim against the ASP. A high level flow diagram is of this is shown in figure 2.2.

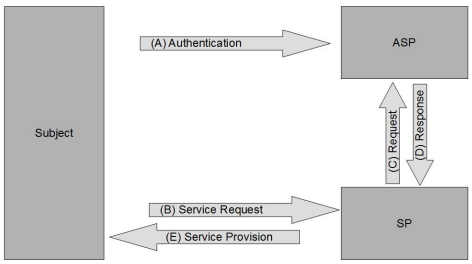


Figure 2.2: High level SSO flow

2.3.3 Platforms and Protocols

XenApp and XenDesktop

XenApp and XenDesktop [42] are two platforms developed by Citrix which provide windows based software as a service (SAAS) or infrastructure as a service (IAAS) respectively. This allows user's to run their virtual applications on a centralised server through a number of different devices, ranging from thin-clients and mobile phones to desktops. The virtual application can be accessed either from a users web browser or through a dedicated Citrix receiver. Traffic between the client and the centralised server is sent over a TLS encrypted channel.

These platforms use a protocol called Independent Computing Architecture (ICA). In contrast to other Virtual Network Computing (VNC) services which uses the Remote Frame Buffer protocol (RFB) to scrape the screen and send it over to the client, ICA sends high level commands to the client similar to Microsofts Remote Desktop Protocol (RDP). This allows XenApp and XenDesktop to work more efficiently in situations where network throughput and reliability is a concern.

Both these platforms allow users to temporarily suspend a session and resume it on a new device. This feature allows users to easy migrate between different physical environments as well as faster re-authentications after a break. In Citrix products this feature is known session roaming.

As XenApp and XenDesktop provides windows based software, the standard authentication protocol is also Kerberos. Default is also a user name-password

based login, but it is possible to change to allow other authentication methods aswell. XenApp is SSO in the sense that one authentication is enough to use all applications provided and XenDesktop in the sense that authentication to the desktop authorises use of all applications on the desktop as well. A more in depth description of the authentication in these platforms can be found in [43].

NetID

NetID is a smart card solution developed by SecMaker. Authentication with NetID is done by inserting the smart card in a card reader and providing a secret PIN-code. NetID is compatible with a large number of different authentication methods and platforms, including Microsoft Windows and Citrix products. The smart card itself contains a user certificate which is used to identify user access and permissions. A more in depth technical description of NetID can be found in [44].

NetID places a lot of focus on SSO, although their perception of SSO, which is described in [45], differs slightly from the description in section 2.3.2. To them SSO is *the fuzzy feeling which takes place when a user can ease access most of the information needed for their work.*

In the default authentication procedure NetID will provide the user with a Kerberos ticket which can provide a more traditional version of SSO, but there is also support for alternatives like e.g. SAML.

FIDO

FIDO is a protocol for "password-less" authentication. It is maintained by the FIDO Alliance and the first version (1.0) [46] was released in December 2014. A big advantage to FIDO is that it allows authentication to be done with whatever technology is available to the subject, e.g. biometric data through face recognition, fingerprints, palm veins etc. or token authentication like RFID, dongle, etc.

There are two protocols defined by FIDO: Universal authentication framework (UAF) and universal second factor (U2F). UAF is used to provide any form of authentication. e.g. a subject registers to a service by choosing and performing any authentication method available to them on their current device. Now they can authenticate themselves on that device by repeating the authentication registered. U2F is used to make authentication stronger by providing multi-factor authentication. A service using U2F requires users to select and perform a secondary mean of authentication on registration. At any time after that, the service can replace the original authentication method with this secondary authentication method.

FIDO works as a key store, i.e. for every service registered, FIDO creates an asymmetric key pair and sends the public key to the service over a TLS secured channel. The private key is stored in the FIDO application and can (only) be unlocked and used by providing the authentication chosen at registration. The private key will then sign a challenge from the service, proving the user is in control of the private key, and thereby authenticating itself.

Chapter 3
Identification of End Users

"Data is the source of invention because it defines the need."

— Karen Holtzblatt (1998) [16]

3.1 Survey

In order to get a better overview of the targeted end-users, contextual interviews were set up. The interviews were scheduled for approximately one hour sessions. This time span of one hour, was determined because most of the relevant data was intermittent, and longer sessions were unlikely to yield better or larger data sets. Instead, interviewees were asked to save or reenact targeted tasks and situations. Following the interviews, interpretation sessions were held. Every interpretation session covered two interviews and resulted in an insight list as well as sequence models of interesting user actions. In total interviews were made with 4 office workers from a corporate environment, 2 nurses from the health care sector and 2 salesmen from retail.

The results are presented in the analysis later in this chapter and in order to make the results more concrete and observable from an outside perspective, three personas, Tom, Mary and Billy, were created, one for each interviewed work role. They are presented in Appendix A. Reading them before the result of the survey can help the reader to interpret its result.

3.2 Sequence Models

3.2.1 Corporate

In total five sequence models were made over observed user behaviours among office workers. Figures 3.1 and 3.2 describe the work sequence when a user wants to log into the internal network, either internally or from an external network. From a user's perspective, both these sequences are longer than they have to be, and although the breakdown at the time of observation didn't seem to have any impact on the user, if these situations are frequent enough, they can accumulate into a breakdown.

Figures 3.3 and 3.4 are the same sequence observed in two different users, the later where a breakdown occurred. Since they occur at a time when the user is just getting started or updated to the task the breakdowns in these sequences help build frustration through repetitive actions, especially Figure 3.4, which requires the user to take multiple steps back. Around these sequences there is a higher risk that users need a break.

In Figure 3.5, a user leaves the workspace to coordinate with a coworker and later returns and resumes working. This sequence shows almost every interaction a user does towards the security system; Locking the screen when leaving the computer, authenticating oneself when returning and the breakdown when being denied.

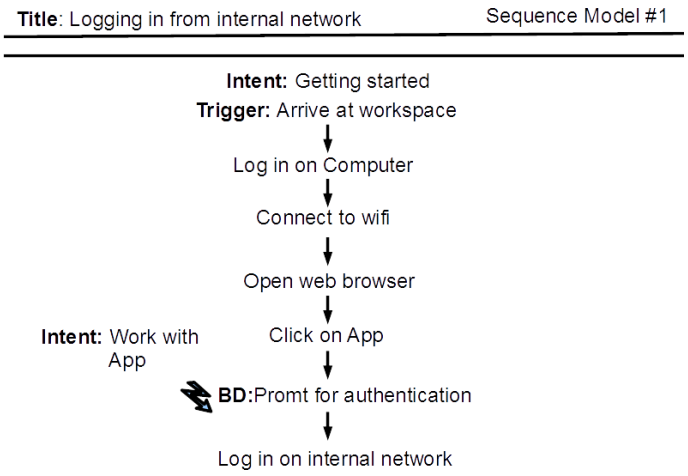


Figure 3.1: Corporate sequence model 1

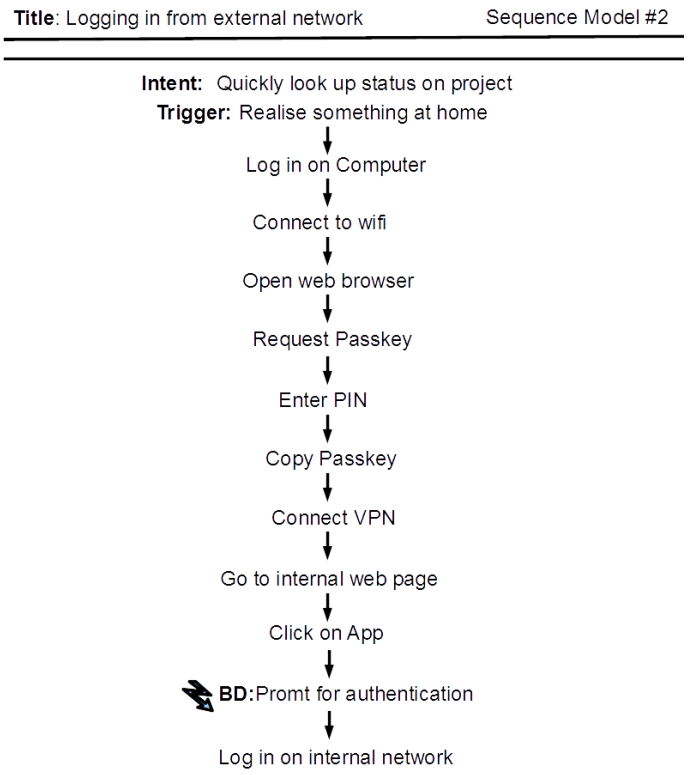


Figure 3.2: Corporate sequence model 2

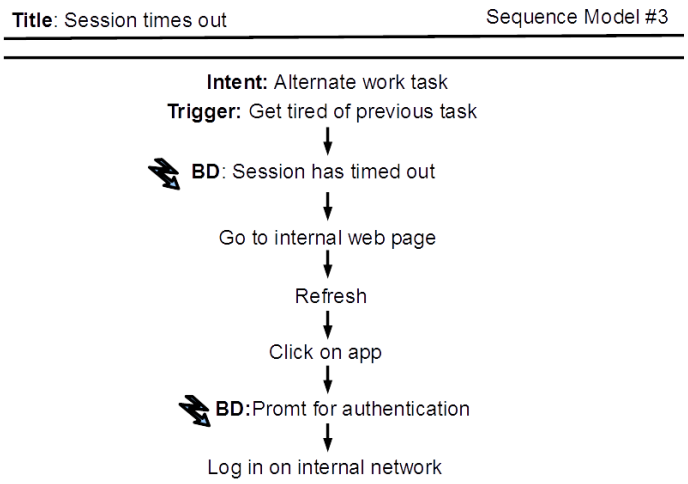


Figure 3.3: Corporate sequence model 3

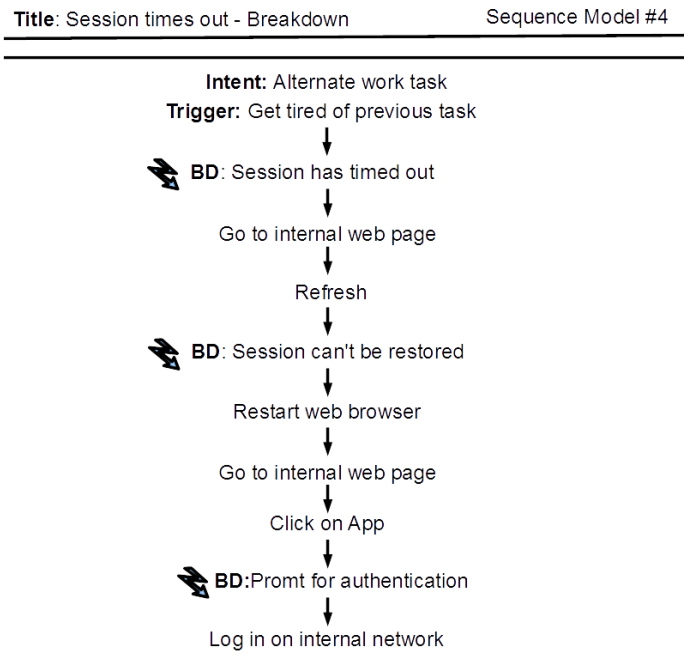


Figure 3.4: Corporate sequence model 4

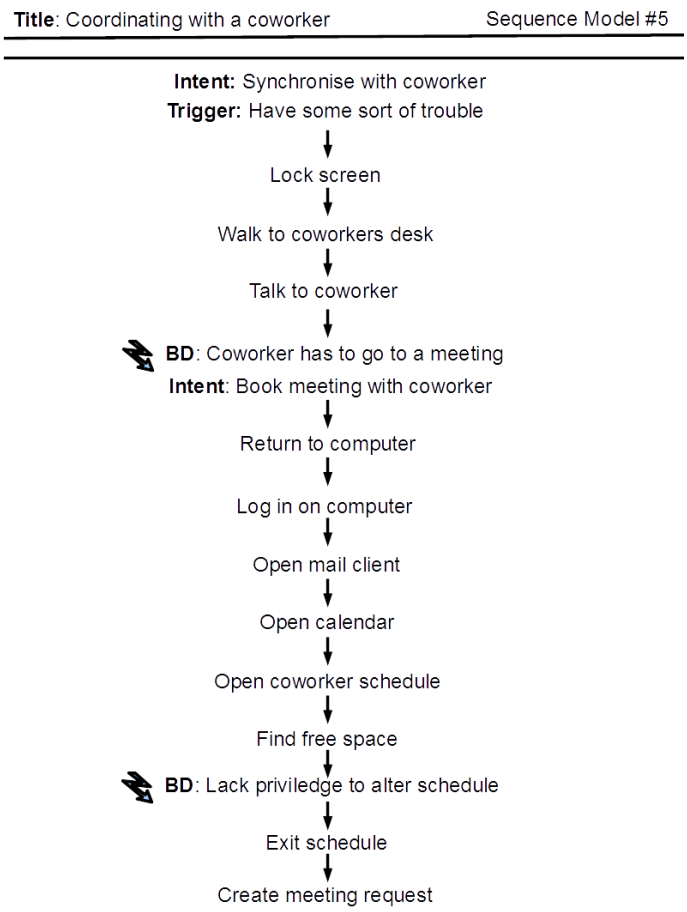


Figure 3.5: Corporate sequence model 5

3.2.2 Health Care

Two sequence models were made from the interviews with nurses. Figure 3.6 describes how distribution of medicine works. According to the interviewed nurses, the breakdown described in the sequence model is not rare. Sometimes the computer freezes twice in a row and if that happens they would switch from the laptop on the tray to a stationary computer further away, making them run back and forth. Figure 3.7 describes the system for notifying municipalities that a patient, who is also a patient in one of the municipality's health care programs, has arrived to the hospital. According to both interviewees, this was the single most loathed interface within the whole health care sector. The reason for this was how unresponsive it was in combination with policies that required all requests to be sent before noon. In both of these sequences we can see that the system must to be responsive end-to-end in order to be secure. If a client's hardware is too slow or buggy, it can lead to security that is worse than no security.

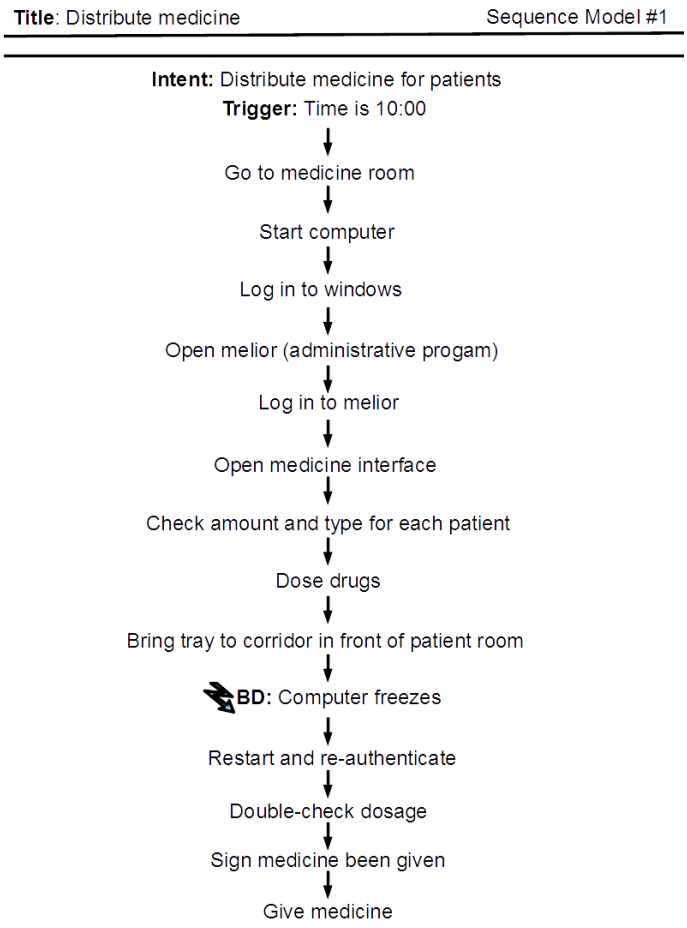


Figure 3.6: Health care sequence model 1

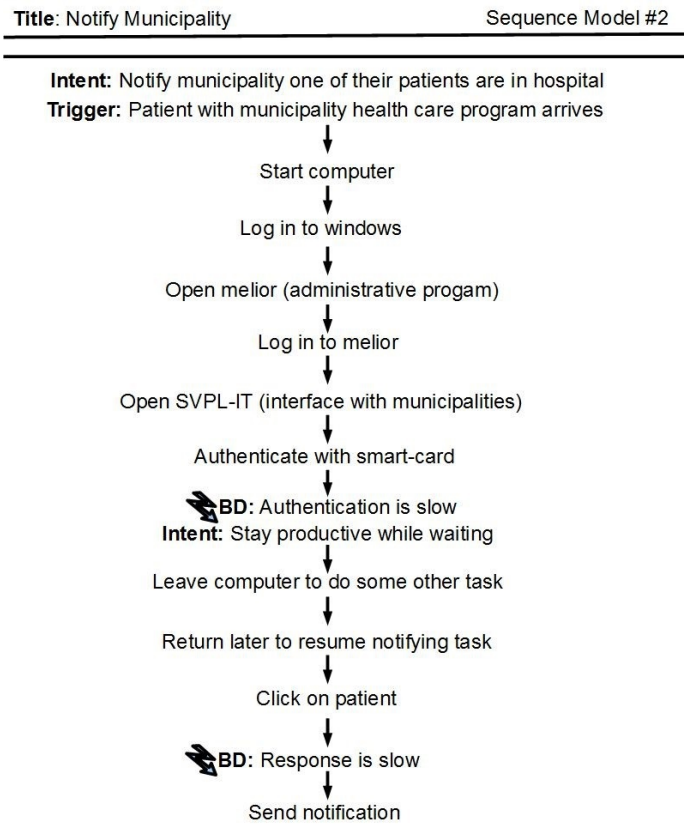


Figure 3.7: Health care sequence model 2

3.2.3 Retail

From the interviews made in retail, three sequence models were drawn. Figures 3.8 and 3.9 describe when a salesman assist a customer with selecting products through a computer interface. In both sequences there is a lot of repetition when the salesman leaves the workspace and comes back. As can be seen in the sequence models, this leads to neglecting to log out in order to avoid these repetitions and save time. Figure 3.10 shows when a customer wants to pay with credit cards. It is interesting to note that salesmen does not only need to authenticate themselves. Sometimes they also need to authenticate their customers.

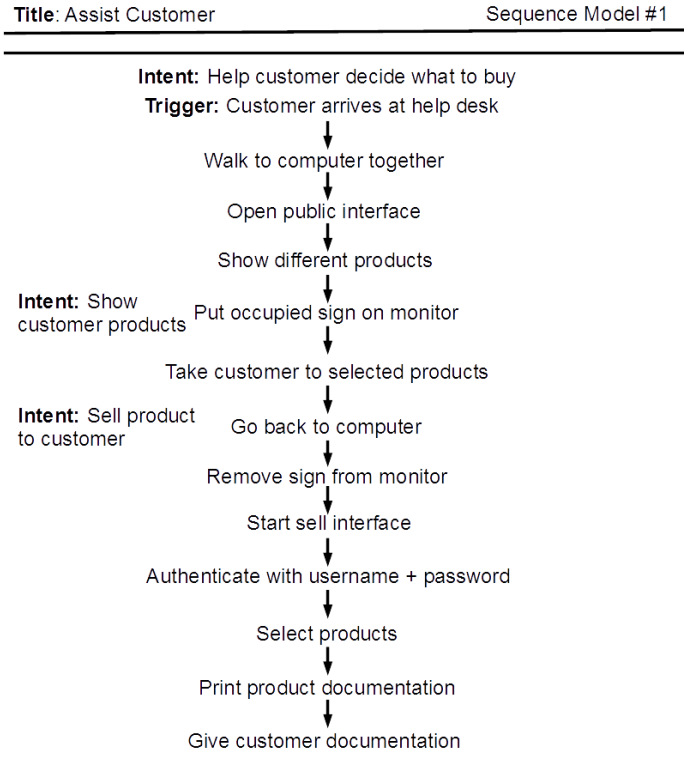


Figure 3.8: Retail sequence model 1

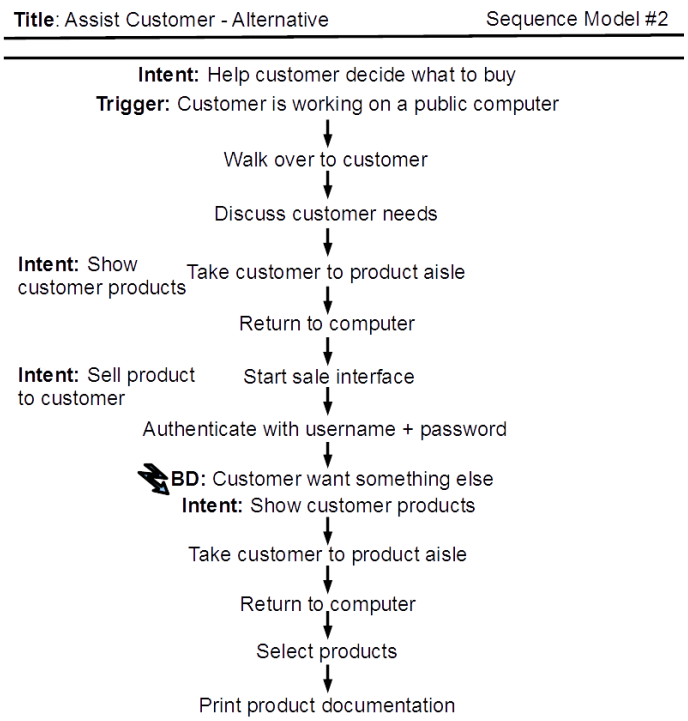


Figure 3.9: Retail sequence model 2

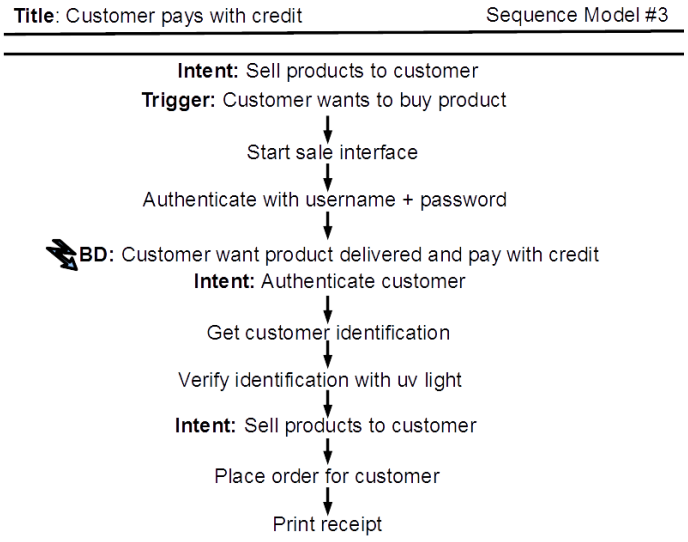


Figure 3.10: Retail sequence model 3

3.3 Analysis

In this analysis, every work role will be analysed based on four common criteria. These criteria are denials, security guidelines, sessions, users and they were picked because during the interviews they were identified to be involved in either the most distracting- or the most insecure behaviour.

Denials are defined as when authorisation fails or when a user for some other reason is denied to perform a specific action. **Security guidelines** are the guidelines a user is informed of by the company, and expected to follow when working with the system. A **session** in this case means one instance when interacting with the system, starting with authentication and ending with a log out or time-out. **Users** are concerning behaviour and preferences observed in the interviewees that focuses on the users, and not the system.

3.3.1 Corporate

Beside the four criteria mentioned above, a fifth interesting observation was made during the interviews with office workers: Copy-paste is used *a lot*.The purpose is mainly transferring data between interfaces, this data is sometimes sensitive information like passwords and the like. Copy-paste is also used for temporarily storing or duplicating information and for re-factoring of information.

Denials are usually accepted without much resistance. One of the interviewed exclaimed "Okay, that didn't work" and then proceeded to try something else. Dialogs were often unread, but they caused a slight focus shift, which prompted users to try a different strategy. At a couple of instances users started experimenting, which was characterised by users flicking between different interfaces and dialogs, and when asked what and why they were doing this one responded with "I'm just looking." During these phases focus shifts were very rapid, approximately once every few seconds. They did not last more than 1-2 minutes, since they are mentally taxing, and often resulted in a work-break. Generally users discarded more than necessary when facing a denial. It seems to be easier to start fresh than to analyse where they went from.

Security guidelines get in the way of the users' preferred way of working. Not a single user obeyed all of their company's security guidelines. For example using the *remembering passwords* functions, or circumventing blocks if they want to listen to music, even though it is not allowed. They will also use coworkers to attain information they might lack privilege of getting if they feel the need it for work.

Multiple **sessions** are running simultaneously, with work done in parallel and tasks alternating. Sometimes a task can be focused on for a longer period of time. When returning to a task it is not uncommon that the session has timed out (see Figures 3.3 and 3.4). These timeouts cause quite severe breakdowns as it requires the users to take a few steps back, and it is not uncommon to lose a bit of progress. When alternating tasks, users seem to re-evaluate their status in the task, i.e. to

read what they have written to make sure everything is still there or to perform other kinds of damage control.

Users like to work on their own devices. This is probably at least partly due to users wishes to be in control. One of the interviewed expressed that they like to use their own devices from home, if they just want to check something really quickly. Another user explained that they like sand-boxing, i.e. isolating themselves in their own environments. If they mess up it will only affect themselves and therefore they can do more experimenting. It is possible that users might put higher demands on their own devices. Denials might not be so unconditionally accepted if they come from your own device.

3.3.2 Health Care

One **denial** is accepted, albeit frustrating. Two denials in short succession is unacceptable and cause users to completely abandon that task. Error messages are not read. They are perceived to be targeted towards someone else, e.g. developers. Interviewees stated that they made that deduction since there was an error code included. This behaviour seems to be learned through their general computer experience and is therefore not likely to be un-learned by new design in the dialogs.

Security guidelines create huge amount of friction. Extreme measures in very *unsafe* ways are taken to work around these policies when they get in the way. The implementation among the different wards are also different, providing friction between wards and hospitals. Signatures to provide non-repudiation are only used when they are required by the system, as it is when dealing and distributing medicine, and never when they are optional, as was the case when noting readings and patient status. Even though all these actions required signatures according to hospital procedures.

Sessions are kept alive for as long as possible since starting new sessions often require two or three different authentication steps. This means leaving the session unattended for long periods of time. If someone arrives to an already authenticated session they will use it directly and not go through the hassle of logging out and authenticate themselves. This is even though they are aware of how sessions are logged and audited, and what consequences it has in regards to patient confidentiality etc.

Some applications require SITHS-card in combination with pin-code to authenticate oneself and while the session is active the SITHS-card need to stay in the dongle. It is not uncommon for nurses to forget the SITHS-card in the dongle, leaving the card in the dongle while going away to do other tasks or finding other SITHS-cards in or around the dongle.

In general, **users** seeem to dislike the computer interface for a few different reasons. First of all, every ward has different needs, but the interface is uniform throughout the whole health care sector. This goes against two of Shneidermann's golden rules, which makes it harder for users to navigate and find the

few functions and selections they need. Secondly, the system is slow and sluggish since the hardware can not handle the size of the system. Lastly, breakdowns in the system usually require users to restart at the beginning. This causes a lot of frustration for obvious reasons.

3.3.3 Retail

Denials were not observed during the interviews, although breakdowns in general caused salesmen to just give up. Since the customer is standing on the side waiting, there seemed to be no time to try again. It is probable that this behaviour would also be applicable to denials.

Security guidelines required salesmen to log out or lock the computers when they left them. During the hour of one of the interviews this policy was broken a total of ten times among the 15 employees at the department. a few times the computer was left alone for long enough that it went into hibernation. During the other interview the salesman was working on a xen-app client (which makes log in and log off quicker), and during the one hour interview this policy was followed, but instead required the salesman to perform authentication approximately every 5 minutes.

When customers ordered an item directly from the salesmen, salesmen were required to properly authenticate customers. They also needed to verify that the customers' identification was not forged by going through a seven step validity check.

The more cumbersome it is to start a **session**, the more reluctant users are to end it. When a session could be frozen with a simple click of a button and resumed with a simple authentication, it was used more than when a session had to be restarted. Other salesmen or customers did not hesitate to hijack a computer with an active session if no one was using it. In fact it was preferred to hijack a session rather than to take an empty computer with an active screen saver on it.

Users prefer the public customer interface instead of their internal system, especially when they are working together with the customer. There seem to be several reasons for this. First of all no authentication is required, secondly the customer interface seem to be more user-friendly. Lastly, when working with the customer, both the customer and the salesman are actively manipulating information on the computer. Customers seem more familiar with the public interface, and the interaction was smoother (although customers also took an active role in the private salex system).

3.4 Requirements

From the analysis, three key requirements on a system were extracted. These requirements provide the basis for design of a invisible security platform.

1. Since users will disregard security guidelines if they become too taxing, it should be made easy for users to be secure, and hard not to be. That is, encourage secure behaviour and provide secure alternatives for the preferred ways of working. Make sure that users know when they break the rules, and audit it, not necessarily for punishment (often disregarded guidelines need to be reevaluated). *The simplest way of working must be secure.*
2. Denial of use should be made more transparent. *A denied action should not trigger users to enter an experimental state of mind.* Instead a denial could trigger a the system to guide users into an alternative permitted sequence, thus making a system denial less binary.
3. The longer it takes to get started, the less willing users are to end sessions, and repetitive actions become more frustrating. *SSO should be available* to avoid these repetitions and *authentication must be quick.* Sessions should be freezable, and restorable with the same or easier means of authentication.

Chapter 4
Design

"When those who benefit are not those who do the work, then the technology is likely to fail or, at least, be subverted."

— Grudin's law [9]

Figure 4.1 contains the overview flowchart of the proposed invisible security platform. A legend for flowcharts can be found in Appendix B. The flowcharts in this section show the systems decision path for arriving at different states and conclusions.

As shown by the flowchart in Figure 4.2, when arriving at the workspace authentication should be proactive, i.e. don't wait for user input to start initiating authentication procedures (e.g. through voice recognition, near field communication (NFC) or facial recognition). But the system should wait for some sort of cue before logging in. If the system has authentication methods that require active user participation (e.g. password or token) the manual inputting of authentication is used as the cue.

Since users should be in control of the interface, even though the system might hold all information needed to authorise user access, waiting for the user to initiate a dialog puts the user in control. It also builds familiarity with the system, which is helpful in making the dialog around denials less abrupt. If the user has a frozen session in the system, it should be immediately resumed to avoid the repetition which makes users less likely to freeze their sessions.

A new session can print a welcome screen. It provides closure for users and gives them a sense of fulfilment. A welcome screen is also a pleasant interaction which builds trust and helps users *like* the system.

There should be a shortcut to freeze a session or logging out with a single step (e.g. a freeze button or grabbing a token). Longer steps make users more prone to skip logging out. When it comes to session timeouts and screen saver, there is a fine line between causing the user frustration by forcibly freezing a session and helping a user maintain security.

Access control when users are authorised is not a problem but when users are lacking authorisation, as shown in figure 4.3, the system should not be outright denying a user. Instead, using a function like the operation pilot, a novel concept

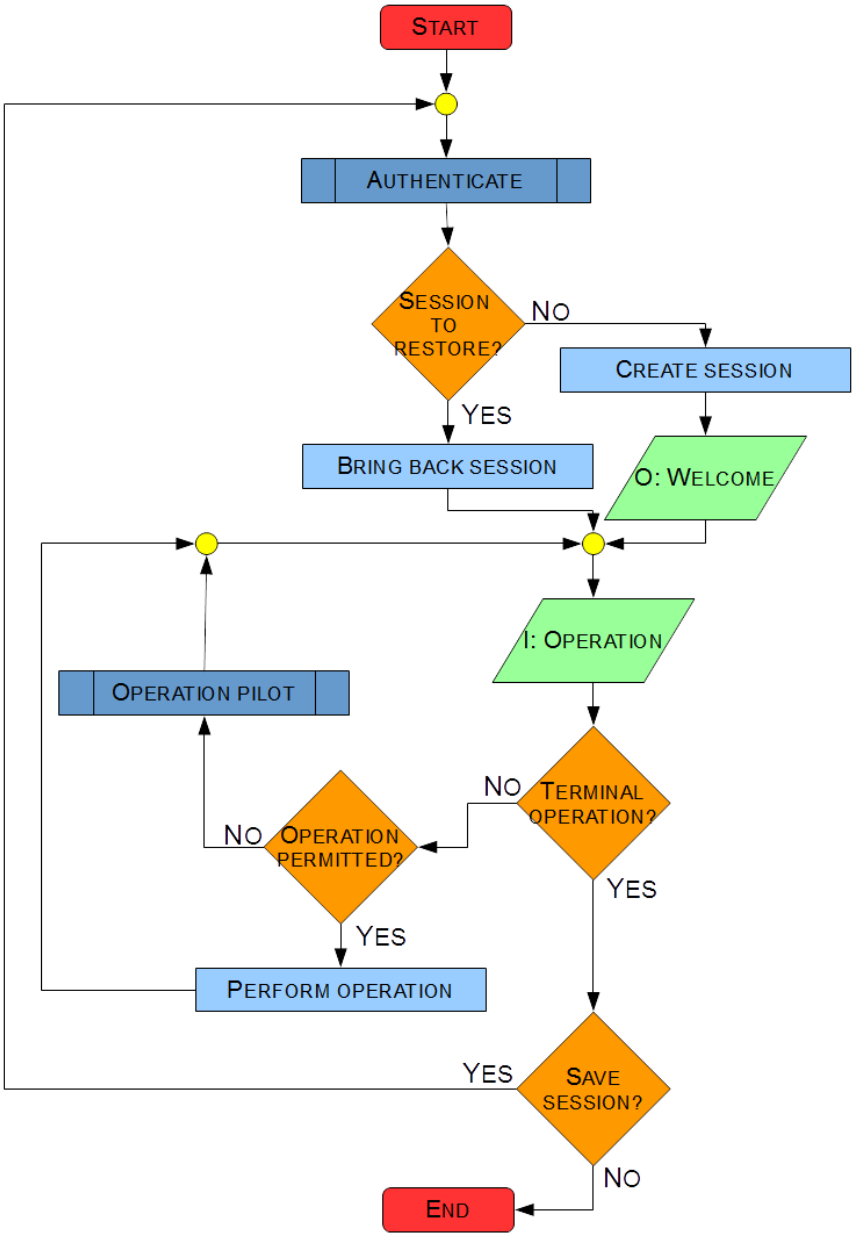


Figure 4.1: System overview flowchart

of this design, which prompts for higher access while allowing users to abort. The operation pilot is less intrusive than an outright denial and places users in control. It should therefore lessen the impact of breakdowns.

In health care, and other situations where denying the operation can have severe consequences, it could be relevant to allow users temporary heightened

access. Then another strategy could be replacing invisible security with "visible security" by displaying eyes or red borders around the screen which makes users aware of the system and that they are being especially monitored (this should be further enforced by extensive auditing e.g. recording the screen and follow-up meetings with supervisors).

Making users aware of monitoring will usually make them behave more obediently, but it also causes more fatigue on the users compared to normal use. Another consequence with this feature is that it allows better layering in the security as information that should only in special situations be available to a user can be put in this layer.

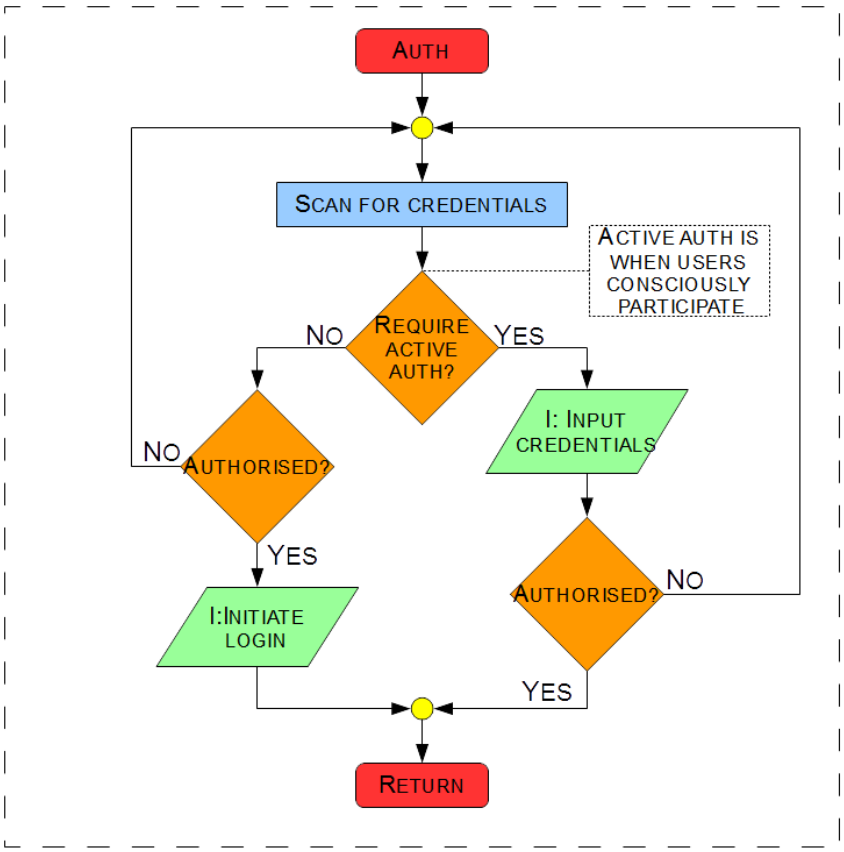


Figure 4.2: Authentication flowchart

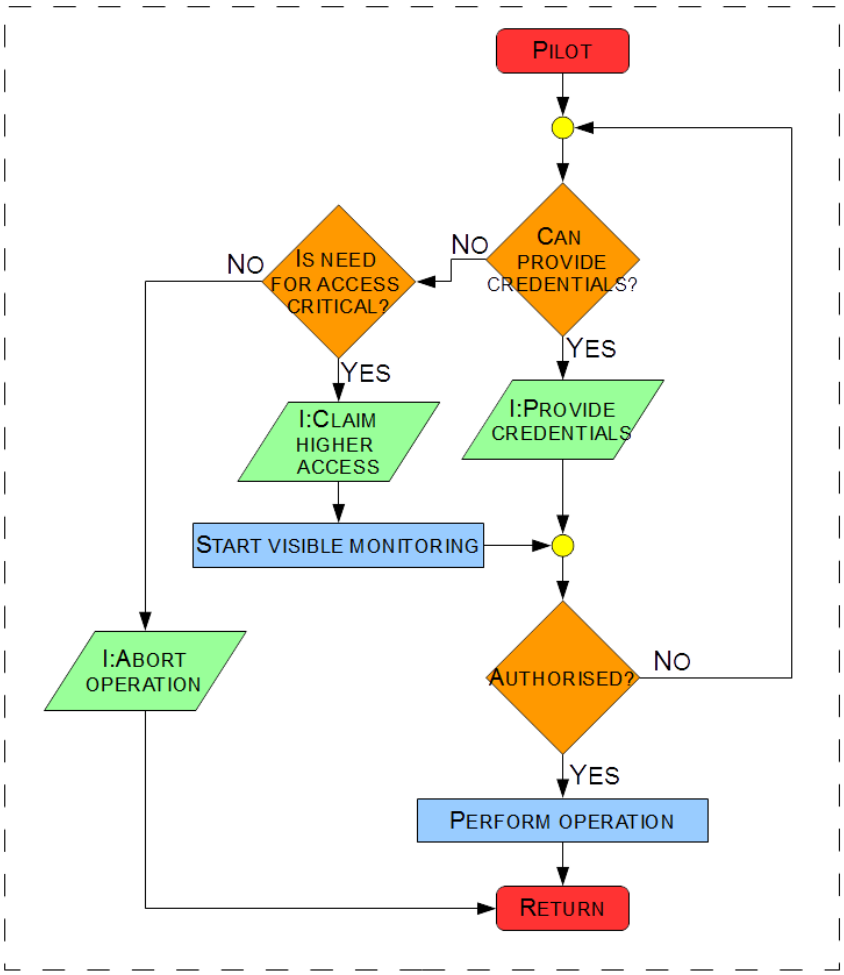


Figure 4.3: Operation pilot flowchart

4.1 Non-functional Requirements

Some of the problems causing the breakdowns observed in the survey were not caused by (lack of) functionality in the system. Rather these breakdowns were caused by the qualitative aspects of it. For example, too long response times and too many options available caused breakdowns among the nurses. These aspects can not be completely addressed in a flowchart, instead we need to look at non-functional requirements (NFRs).

NFRs, or quality requirements as they are called in [47], are the requirements on a system which does not decide what functions the system should provide. Instead these are used to assure the system hold a high enough standard with regard to reliability, security, accuracy, safety, performance, and so on. NFRs should always be either measurable or restrictive. Restrictive NFRs are formulated to express constraints on the system. For example *A user should not be able to obtain access to data without first authenticating herself.*

For invisible security compliant platforms formulating an NFR as *breakdowns need to be kept at a minimum,* although true, NFRs formulated in this way are hard for designers to work with since they are imprecise and unverifiable. A better way is to formulate the NFR as *a user should experience breakdowns on average less than every 10 minutes.*

The process of selecting values for these measurements is not easy, and is often based on the gut feeling of those who develop the NFRs. Although, more systematic approaches exist, among them is the Quality Performance (QUPER) model [48, 49].

The QUPER model gives a linear graph including benchmarks for varying performance in relation to expectations and competition. Following is a rough attempt at defining NFRs for invisible security using the QUPER model.The values used are estimations based on personal observations and influenced by [50] regarding reaction times, and [51] regarding authentication failure rates.

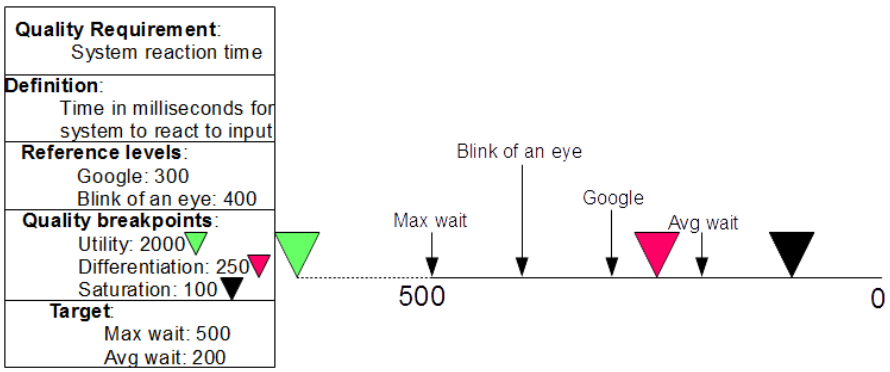


Figure 4.4: NFR: Reaction time

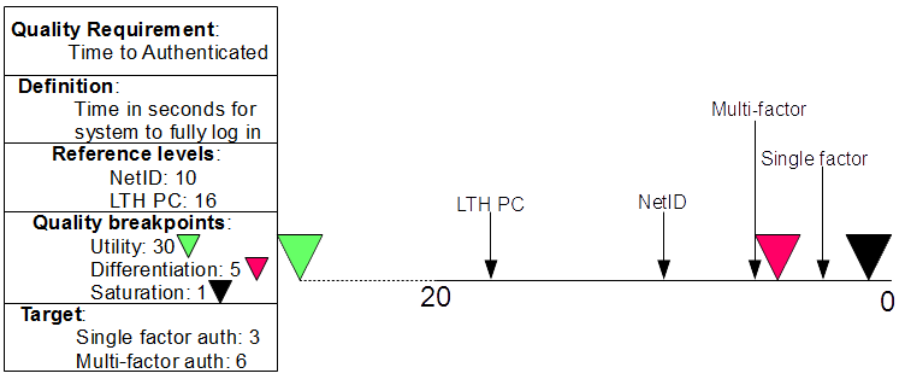


Figure 4.5: NFR: Authentication time

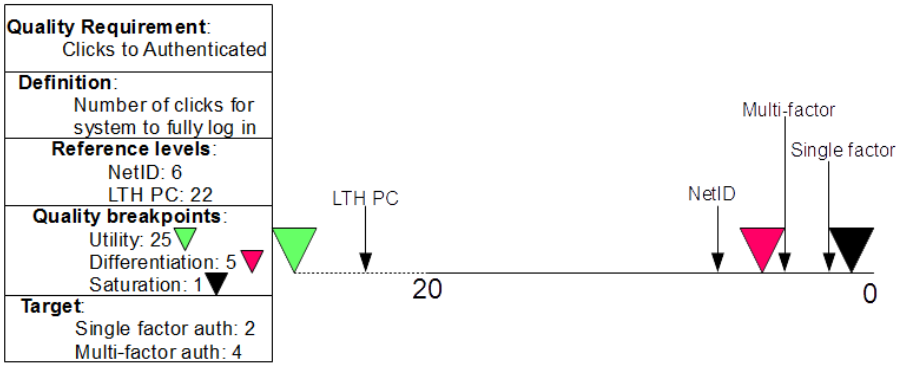


Figure 4.6: NFR: Authentication clicks

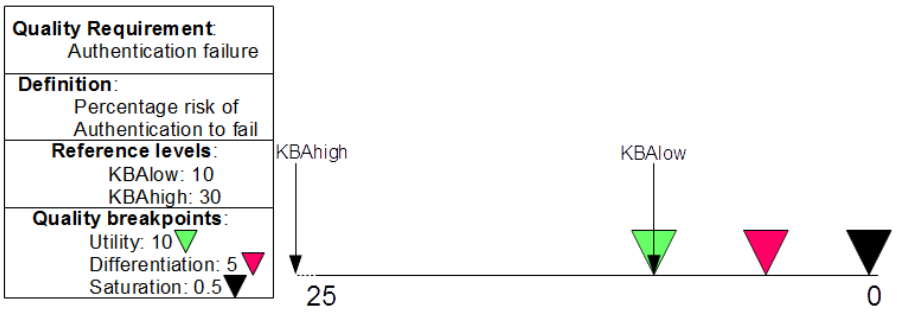


Figure 4.7: NFR: Failure rate

4.2 Comparisons

NetID

Secmaker claims in [52] that users forgetting or leaving their cards is a myth using a comparison between smart cards for authentication and bank cards. The survey in this thesis disagrees with SecMaker's claim. As has been described earlier in the thesis, this problem seems to be twofold.

First, there is an attitude issue. Ordinary users do not think about security when they are working. Their intent is most certainly that they will return at a later time and resume working on the computer, there is no such plan when withdrawing money from an Automated Teller Machine (ATM), which makes a comparison between the two scenarios unfair. When users are leaving their workspace they will perform a risk assessment. It is obvious from the contextual interviews that leaving a session open is not seen as a high risk. It is also improbable that users can be convinced of the risks. The best way to solve the attitude issue is probably through physical constraints; e.g. attaching the smart card to the person with a wire. This would make it harder for a user to leave the session running than to simply take the card and log out.

The second issue is performance related. When system performance falters, breakdowns occur, and, as describe earlier, these breakdowns lead to unsecure behaviour among users since it triggers users to look for new solutions to their objective. For a smart card solution to be viable, it should conform to the NFRs described in this chapter.

The reference values in Figures 4.5 and 4.6 for NetID is much lower than the LTH school computers. The reason for this is that LTH computers use a username-password combination for authentication while NetID use token with four-digit pincode combination. The reference values for NetID are good, but they still fall a bit short of the target levels for invisible security. The reason for this is that NetID waits for users to initiate before starting logging in, in combination with the manual dexterity required to insert the smart-card into the dongle.

Fido

Fido has taken a quite flexible approach to authentication, as it allows users to select which method of authentication they want among the ones currently available. This is a good way to get users to like the system as it places them in control of the situation. It also allows users to discriminate against the methods of authentication they dislike. It also allows users to update their authentication procedures. This is especially relevant in biometric authentication, where algoritms, sensors and other hardware advances are still being improved.

There are however security concerns. Users do not have a habit of choosing the most secure options and they tend to be reluctant to upgrade and abandon old methods.

The idea behind U2F is hard to make compliant with invisible security. The fact that it is up to the service to choose when to require second factor authentication makes it likely to cause breakdowns. Furthermore, the second factor of

authentication is not normally prompted for. This makes users more aware and it is likely to make them more flustered and therefore more prone to breakdowns if authentication fails for some reason.

XenApp and XenDesktop

Citrix's platforms provides session roaming very similar to the suggested method in this invisible security design. This feature provides users with a great deal of mobility as they can freeze their session and then resume it on a different device. Salesmen can move their session as they are moving with the customers, and they will not freeze sessions for other salesmen when they freeze the session, as is the case with many of Microsoft's Windows products today.

Another nice addition with these solutions is that they allow for BYOD. Users can use any device they want as long as they either have a Citrix receiver application or a web-browser. This allows users to work distributedly or to make quick checkups from their own devices. It also provides better security as user's data is kept in a data center instead of on users own devices. Therefore, responsibility of security is mostly moved from users to system administrators, who are better equipped and trained to handle it.

SSO

SSO differs in definition between the platforms. It seems that with the recent popularity rise of SSO as a buzzword, *sales talk* is a factor. SecMaker's definition in [45] is actually resembling the idea behind invisible security of keeping the system from distracting users quite a lot; *the fuzzy feeling which takes place when a user can ease access most of the information needed for their work.* In their vision they also raise the point that application providers need to comply with the demands to make their applications SSO compliant, since most platforms already have support for SSO solutions built-in.

SSO is a fundamental requirement to invisible security for the very reason it has risen in popularity. It allows users to avoid unnecessary repetition of authenticating themselves, which is both frustrating and time consuming.

Denials

Denials in the invisible security design are handled with a very novel approach in the operation pilot. The possibility to temporarily raise a user's access with trade-off of harsher accounting, makes for interesting possibilities. None of the platforms reviewed for this thesis provide anything resembling this. Their authentication processes are all binary – either you are authorised or you are not.

Invisible security also aims at helping users to obtain authorisation. If they lack the authorisation needed, the system should attempt to guide users through the process or point to someone who can help. Again, this line of thinking is novel with invisible security and lacking in the other platforms.

Chapter 5
Discussion

"Essentially,
all models are wrong, but some are useful."

— George E. P. Box (1987) [53]

The overall purpose of this thesis was to design a concept of a responsive enterprise security platform. In order to do this, a survey of target users was conducted using contextual interviews.

Contextual interviews are not very well suited for making NFR specifications. More specifically, contextual interviews are aimed at gathering design data which can identify patterns and reasons for user behaviour. NFR on the other hand typically requires hard data. As such, the NFRs provided in this report are rough estimates and require more statistics and data to gathered before any major conclusions can be drawn from them. As of now, they mostly provide an rough indication of the NFRs around invisible security.

Another problem with contextual interviews is that they assume a working product for users to work with during the interview. Therefore contextual interviews are harder to utilise when designing brand new systems. This also makes them less likely to lead to any revolutionary changes and ideas. Although, for the purpose of this thesis, the use of contextual interviews felt relevant and effective to make good estimations of user needs and behaviour, and the sequence diagrams were helpful in identifying user patterns and underlying reasons behind that behaviour.

Research Questions

- *How do you design a platform which handles security without distracting users from their main work-tasks?*

The design chapter in this thesis makes an attempt at answering this question. The design decisions are based on the results from the user survey, but they have yet to be tested and fully evaluated. Therefore the suggestions are still purely theoretical.

One identified risk is with the operation pilot. The operation pilot has the goal of making denials less binary and as such, making users less flustered when

experiencing a denial. There is a risk that this will not work as intended and instead make users waste their time while trying to get authorisation to perform an operation they are not under any circumstances allowed to perform. This is likely to build up frustration among users and make the inevitable breakdown even more severe. In situations like those, maybe denying a user is a necessary evil.

A interesting note for developers to remember is that error messages are not read. So, important information should not be put in error messages. One of the interviewees made an interesting comment on this subject: "Error codes are for developers, this message has an error code, therefore this message is for developers and not for me." Following this logic, make sure to allow the information to be accessible from other through other means as well, even after a user has gotten rid of the pop-up dialog since users are likely to just want it to disappear as quickly as possible. This line of thinking is also consistent with Shneiderman's golden rule of allowing easy reversal of action.

To avoid breakdowns and frustration, it is important to not only look at design and functionality. Performance is an important aspect which need to work seamlessly end-to-end. In this thesis four important NFRs have been addressed, which need to hold a high standard in order to maintain an unobtrusive user environment. Authentication failure rate and system reaction time were during observations determined to cause the most severe breakdowns and frustration among users. But long authentication procedures where one of the main causes that kept users from logging out when they left the workspace unattended, and this was the largest observed security vulnerability. Therefore all four of these NFRs are essential to an invisible security platform.

- *Is it possible to retain security while lowering user interaction?*

One could argue that lowering user interaction actually raises security. As shown in the survey of users, they are normally not in a mindset that promotes secure thinking. Having security depend on responsible user interaction makes the attack surface larger. Confining user interaction with security into special areas in access control allows system designers to help users think about security only when it is needed, so they do not have to worry about it all the time.

Traditionally, security and usability have been seen as two ends of a line. The widespread use of passwords as means of authentication can be a large reason for this line of thinking, since passwords are either considered hard to create and remember when security concerns are address or easy to guess and crack when user's needs are focused. This thesis has made an attempt at looking at this problem from a different angle. Working with the basis that security needs to take precedence over usability or usability over security is outdated. Many reports highlight that users' mistakes is the most common factor in security breaches. Therefore, it is assumed in this thesis that usability is in fact included in security. A user friendly platform is more secure than a system that causes users frustration and forces them to make focus shifts between work and security on a constant basis.

- *Which decisions can be automated, to what extent and in what situations?*

There are methods to perform access control authentication in an automated manner. Some biometric authentication, like facial or voice recognition, can work automatically from a distance. Token authentication can also automate authentication, e.g. like Intel, who in [5] describe their vision of invisible security using wearable computing. When these authentication methods discover a user they can prepare for the user to login, allocate resources for their session and prepare the session to be ready. This would lower the authentication time described in Figure 4.5. The final decision to log in should always be left to the user. Otherwise user control is forfeit which will cause many users distress.

A similar feature can be utilised when users are leaving the workspace, if a user leaves the workspace the session can be automatically frozen so that other users will not be able to use the session. It is important that sessions are not left unattended since, as observed during the user survey, other people are more likely to hijack a running session rather than start their own.

Another security feature that can be automated but has been outside the scope of this thesis is tracking of user behaviour. When users are interacting with computers, their strategy form a pattern. Things like writing speed and paths to get into specific functions can be tracked automatically. As long as this behaviour is consistent with past experiences, we can be fairly sure that the user is honest and in fact who they say they are. But if these behaviours start to deviate from normal behaviour, it is an indicator that something is wrong and we might need to lower user permission or request stronger authentication to make sure the user is not an impostor.

The last and most common thing to automate is the repetition of authentication in a single sign-on (SSO) scheme. SSO is very effective in lessening breakdowns in users work-flow. In a corporate environment there are also less privacy concerns to consider, so in theory there should be no problems in utilising SSO without first notifying the user. Although, a security concern is that if a user loses their session to a malicious intruder, the impact can be far more severe.

- *How are users' trust and mental model affected when user interaction is reduced?*

From a user's perspective there are certain dangers when lowering user interactions. As described in Chapter 2, automation does not supplant human interaction. It is important that user expectations and trust match the system capabilities. Therefore the system should keep a dialog with users to build trust and assist them by providing mental models at a level suitable for them. Things like allowing users the final say before logging in and providing a welcome screen when they arrive the first time is a good method of building trust and it also helps in calibrating the resolution of automation. For highly automated systems, mimicking human-human interaction seems to be effective for calibrating resolution, since we already obey to the media equation.

Since the interaction removed in SSO-type solutions is repetition of interaction already made, the risks in user trust involved is fairly small. Although, if applications are very different, e.g. if users are mixing private applications with

work applications there is a risk that users feel the system is leaking their private information. I.e. telling applications about things the user do not want that application to know.

There is a risk that users will feel that automatically freezing a session when they leave the workstation is annoying, since it makes users repeat authentication more often, this can be solved by automatically resuming the session if users return within a specified period of time and no one else has been on that mainframe.

If tracking of user behaviour is used as a form of authentication, there are a number of design decisions that need to be considered from a HAC point of view. Taking another page from human-human interaction; be conservative how and when telling users that they are being tracked. Making users aware they are being observed, like in the case of *visible security*, when temporarily heightening access, tends to make users self-conscious. This is generally bad for productivity and raises fatigue of work. The same is true when tracking user behaviour.

5.1 Future Work

For this thesis a survey of user behaviour has been made. The survey resulted in a design suggestion of an invisible security platform. The next step would be to test, implement, and evaluate this model. More tests and investigation also need to be made in order to obtain better measurements for the NFRs.

Another comparative survey of different authentication methods within invisible security would also be helpful for system designers who wish to use the findings in this thesis.

If system designers for some reason want to keep knowledge based authentication, new methods need to be developed since passwords and pin-codes are neither user friendly nor secure. This thesis has worked with the hypothesis that the problems in passwords and pin-codes are fundamental flaws in knowledge-based authentication. But it is possible that this hypothesis is incorrect. Maybe there are ways to create user friendly and secure knowledge based authentication schemes. In the words of Donald Norman: Representation fit for human use makes us smarter. And it is obvious from years of use that passwords are not a good fit.

Chapter 6
Conclusions

In this master's thesis a survey of users has been conducted using contextual interviews. The survey concluded that when users are confronted in a conflict between company enforces security guidelines and their planned method of conducting work, security guidelines are neglected. This is amplified when the system performance is poor and when guidelines consist of frequent repetitive actions like long or complicated authentication procedures. Furthermore denial of access or permission to perform certain operations were determined to be highly disruptive to user's work. Based on this three requirements for an invisible security platform were formulated. These requirements consist of:

1. The simplest way of working must be secure.
2. A denied action should not trigger users to enter an experimental state of mind.
3. Authentication must be quick, SSO should be available to avoid repetitions.

Based on the result from the user survey, a design solution has been proposed including a proactive authentication method with SSO, session handling resembling session roaming in Citrix solutions and an operation pilot for guiding users when permission is not immediately granted. The design solution also address the need for a certain performance in order to keep users from getting distracted. Authentication should always leave the final decision to the user in order to keep users in control of the interface. Session roaming allows for fast re-authentication of users who work in a mobile environment, it also eases BYOD and sharing workspaces. The operation pilot was designed to allow temporary heightened access with the trade-off of more a extensive, and noticeable accounting procedure called visible security. This design has yet to be tested.

References

[1] Murdoch, S., Anderson, R., and Sion, R., eds. *Verified by Visa and MasterCard SecureCode: Or, How Not to Design Authentication.* University of Cambridge, Computer Laboratory, Cambridge, UK, 2010.

[2] Verizon. *2014 Data Breach Investigations Report*. 2014. URL: http://www.verizonenterprise.com/DBIR/2014/ (visited on 02/17/2015).

[3] Verizon. *2013 Data Breach Investigations Report*. 2013. URL: http://www.verizonenterprise.com/resources/reports/rp_data-breach-investigations-report-2013_en_xg.pdf (visited on 02/18/2015).

[4] Splashdata. *"123456" Maintains the Top Spot on SplashData's Annual "Worst Passwords" List*. 2015. URL: http://splashdata.com/press/worst-passwords-of-2014.htm (visited on 02/18/2015).

[5] Nagisetty, R and Booth, C. "THE NEW PARADIGM: WHEN AUTHENTICATION BECOMES INVISIBLE TO THE USER." In: *Intel Technology Journal* 18.4 (2014), pp. 198 –209. ISSN: 1535864X.

[6] Pilloud, C. et al. *Commentary on the Additional Protocols: of 8 June 1977 to the Geneva Conventions of 12 August 1949*. Martinus Nijhoff Publishers, 1987.

[7] Hewett, T et al. *ACM SIGCHI curricula for human-computer interaction.* New York : Association for Computing Machinery, cop. 1992. ISBN: 0897914740.

[8] Norman, D. *The design of everyday things.* New York : Basic Books, 2002, 2002. ISBN: 0465067107.

[9] Norman, D. *Things that make us smart : defending human attributes in the age of the machine.* Reading : Addison-Wesley, cop. 1993, 1993. ISBN: 0201581299.

[10] Shneiderman, B. and Plaisant, C. *Designing the user interface : strategies for effective human-computer interaction.* Harlow : Pearson Education, cop. 2014, 2014. ISBN: 9781292023908.

[11] Cooper, A. *The inmates are running the asylum : [why high-tech products drive us crazy and how to restore the sanity].* Indianapolis, Ind. : Sams, cop. 2004, 2004. ISBN: 0672326140.

[12] DIS, ISO. *Ergonomics of human system interaction-Part 210: Human-centred design for interactive systems*. ISO 9241-210:2010.

[13] Norman, D. "Human-centered design considered harmful." In: *Interactions* 12.4 (2005), pp. 14 –19.

[14] Dearden, A. "User-Centered Design Considered Harmful (with apologies to Edsger Dijkstra, Niklaus Wirth, and Don Norman)." In: *Information Technologies & International Development*. Vol. 4. 3. 2008, pp. 7 –12.

[15] Abras, C., Maloney-Krichmar, D, and Preece, J. "User-centered design". In: *Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications* 37.4 (2004), pp. 445–456.

[16] Beyer, H. and Holtzblatt, K. *Contextual design : defining customer-centered systems.* San Francisco, Calif. : Morgan Kaufmann Publishers, cop. 1998, 1998. ISBN: 1558604111.

[17] Adlin, T and Pruitt, J. *The Essential Persona Lifecycle. Your Guide to Building and Using Personas.* Burlington : Elsevier Science, 2010., 2010. ISBN: 9780123814197.

[18] Holtzblatt, K, Wendell, J, and Wood, S. *Rapid contextual design. a how-to guide to key techniques for user-centered design.* Morgan Kaufmann series in interactive technologies. Burlington : Elsevier, 2005., 2005. ISBN: 0080515711.

[19] Carroll, J. *HCI Models, Theories, and Frameworks : Toward a Multidisciplinary Science.* The Morgan Kaufmann Series in Interactive Technologies. Morgan Kaufmann, 2003. ISBN: 9781558608085.

[20] Reeves, B. and Nass, C. *The media equation : how people treat computers, television, and new media like real people and places.* CSLI Publications. Cambridge : Cambridge Univ. Press, 1996, 1996. ISBN: 157586052X.

[21] Parasuraman, R. and Riley, V. "Humans and automation: use, misuse, disuse, abuse." In: *Human Factors* 39.2 (1997), pp. 230 –253.

[22] Christoffersen, K. and Woods, D. "How to make automated systems team players". In: *Advances in human performance and cognitive engineering research* 2 (2002), pp. 1–12.

[23] Hancock, P. "Automation: How much is too much?." In: *Ergonomics* 57.3 (2014), pp. 449 –454.

[24] Norman, D. *The 'problem' with automation: inappropriate feedback and interaction, not 'over-automation'*. Oxford University Press, 1990. ISBN: 9780198521914.

[25] Bradshaw, J. et al. "The Seven Deadly Myths of 'Autonomous Systems'." In: *IEEE Intelligent Systems* 28.3 (2013), pp. 54 –61.

[26] Federal Aviation Administration. *Human Factors Design Standard*. 2007. URL: `http://hf.tc.faa.gov/hfds/download.htm` (visited on 04/01/2015).

[27] Sheridan, T., Parasuraman, R., and Wickens, C. "A model for types and levels of human interaction with automation." In: *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans* 30.3 (2000), pp. 286–297. ISSN: 10834427.

[28] Bailey, N. and Scerbo, M. "Automation-induced complacency for monitoring highly reliable systems: the role of task complexity, system experience, and operator trust." In: *Theoretical Issues in Ergonomics Science* 8.4 (2007), pp. 321 –348. ISSN: 1463922X.

[29] Culley, K. and Madhavan, P. "Trust in automation and automation designers: Implications for HCI and HMI." In: *Computers in Human Behavior* 29.6 (2013), pp. 2208 –2210. ISSN: 0747-5632.

[30] Lee, J. and See, K. "Trust in Automation: Designing for Appropriate Reliance." In: *Human Factors* 46.1 (2004), pp. 50 –80. ISSN: 0018-7208.

[31] Dzindolet, M. et al. "The role of trust in automation reliance." In: *International Journal of Human - Computer Studies* 58.Trust and Technology (2003), pp. 697 –718. ISSN: 1071-5819.

[32] Lee, J. "Review of a pivotal human factors article: 'humans and automation: use, misuse, disuse, abuse'." In: *Human Factors* 50.3 (2008), pp. 404 –410.

[33] Jamson, A.H. et al. "Behavioural changes in drivers experiencing highly-automated vehicle control in varying traffic conditions." In: *Transportation Research, Part C: Emerging Technologies* 30 (2013), pp. 116 –125.

[34] Kircher, K., Larsson, A., and Hultgren, J.A. "Tactical Driving Behavior With Different Levels of Automation." In: *IEEE Transactions on Intelligent Transportation Systems* 15.1 (2014), pp. 158 –167.

[35] Hoffman, R., Bradshaw, J., and Hawley, J. "Myths of automation, part 2: Some very human consequences." In: *IEEE Intelligent Systems* 29.2 (2014), pp. 82–85. ISSN: 15411672.

[36] Karp, A., Haury, H., and Davis, M. "From ABAC to ZBAC: the evolution of access control models". In: *Proceedings of the 5th International Conference on Information Warfare and Security, ed. EL Armistead*. 2010, pp. 202–211.

[37] Tcsec, DOD. "Trusted computer system evaluation criteria". In: *DoD 5200.28-STD* 83 (1985).

[38] Ferraiolo, D. et al. "Proposed NIST standard for role-based access control". In: *ACM Transactions on Information and System Security (TISSEC)* 4.3 (2001), pp. 224–274.

[39] Gollman, D. *Computer Security*. Chichester, West Sussex, U.K. ; Hoboken, N.J. : Wiley, cop. 2011, 2011. ISBN: 9780470741153.

[40] Vollbrecht, J et al. "AAA authorization framework". In: (2000).

[41] Pashalidis, A and Mitchell, C. "A taxonomy of single sign-on systems". In: *Information security and privacy*. Springer. 2003, pp. 249–264.

[42] Citrix. *Reviewer's guide: XenDesktop 7.6*. 2014. URL: `http://www.citrix.com/content/dam/citrix/en_us/documents/products-solutions/xendesktop-reviewers-guide.pdf` (visited on 05/25/2015).

[43] Kaushal, L. *Authentication on XenApp & XenDesktop*. 2010.

[44] SecMaker. *Net iD Enterprise Technical Description v6.1*. 2013. URL: `http://www.secmaker.se/wp-content/uploads/2014/02/NetiD_Enterprise_Technical_Description_v6.1.pdf` (visited on 05/26/2015).

[45] SecMaker. *Single Sign On, SSO*. 2015. URL: `http://www.secmaker.se/teknikresurser/pki/inloggning-och-sso/` (visited on 05/26/2015).

[46] FIDO Alliance. *FIDO 1.0 specification*. 2014. URL: `http://fidoalliance.org/specifications/download/` (visited on 02/12/2015).

[47] Lauesen, Soren. *Software requirements : styles and techniques.* Harlow : Addison-Wesley, 2002, 2002. ISBN: 0201745704.

[48] Berntsson Svensson, R and Regnell, B. "A Case Study Evaluation of the Guideline-Supported QUPER Model for Elicitation of Quality Requirements." In: *Requirements Engineering: Foundation for Software Quality 21st International Working Conference, REFSQ 2015, Essen, Germany, March 23-26, 2015. Proceedings* (2015), p. 230. ISSN: 9783319161006.

[49] Regnell, B, Berntsson Svensson, R, and Olsson, T. "Supporting Roadmapping of Quality Requirements." In: *IEEE Software* 25.2 (2008), p. 42. ISSN: 07407459.

[50] Lorh, S. *For Impatient Web Users, an Eye Blink Is Just Too Long to Wait*. 2012.

[51] Litan, A. *Knowledge Based Authentication Breached Big Time! Another dagger for Obamacare, the Banks and many others*. 2013.

[52] SecMaker. *Tio myter om smarta kort*. 2015. URL: `http://library.secmaker.com/uploads/accounts/786/54e34aeb8d7b6.pdf` (visited on 05/26/2015).

[53] Box, G. and Draper, N. *Empirical model-building and response surfaces.* Wiley series in probability and mathematical statistics. New York : Wiley, cop. 1987, 1987. ISBN: 0471810339.

[54] reynermedia. *Men at work*. 2013. URL: `www.flickr.com/photos/89228431@N06/11220929004` (visited on 04/23/2015).

[55] Jones, A. *Nurse in Pinar del Rio*. 2006. URL: `www.flickr.com/photos/adam_jones/3793861141` (visited on 04/23/2015).

[56] Lonien, W. *7dbpa114374-camera-salesman*. 2011. URL: `www.flickr.com/photos/wjlonien/6234745438` (visited on 04/23/2015).

Appendix A
Personas

A.1 Tom

Tom works at a Swedish company in Malmö. He has only been there for six months. Toms current work task is to write user manuals for their product line. For that Tom uses word since the company already have finished templates he has to follow. When Tom works he likes to shut himself in a bubble, listening to music helps him concentrate at the task at hand. Tom usually first prints the specification on paper and highlights changes with a marker on the paper at the same time as he writes it on the computer. He usually keeps two word documents open at once, one where he saves information so he can quickly copy and paste it into the other. Once he finished he sends a copy to his supervisor, Vivianne through their mail client. She will proof read it and comment with corrections, which is mark-upped with words' internal comment system.

Tom is quite social. He often joins his coworker for after work and go for lunch together. Besides work he plays pick-up soccer in the weekends and he is a regular at the local Cinema.

Tom [54]

A typical workday for Tom starts at 8 am when he arrives at work. Usually he starts his laptop and while it boots he goes to grab a cup of coffee. When he comes back he logs in with his password. Then he starts his mail client and reads new mails while drinking coffee. When he finished his cup, he plugs in his headphones, logs in to the internal network and starts working "for real". Usually Tom works with preparatory work, first draft and manual correction for 2-3 products at the same time, alternating when he gets stuck or after about half an hour. At those moments he often goes for a stretch, talks a bit with coworkers or refills his coffee.

A.1.1 Tom's Goals

- I want to learn how this company's way of working, and
- work in the development team, and maybe at a later stage
- work as a manager.

A.1.2 What Do Tom Want From Us

Focus on his work. Sometimes Tom gets *in the zone*, and when he gets interrupted he find it hard to get back in it.

Quick and easy authentication. When Tom arrives at the workstation he wants to start working immediately. Since he often ponders on wording while away, it is important to write them down before he forgets them.

Know what's going on. Since he has manager ambitions, Tom still wants to know what happens, learn how everything is connected and who is responsible for what.

A.2 Mary

Mary is a certified nurse working at Scania University Hospital in Lund. She is working in the infection ward which is split into two parts; A and B. Both section has room for 8 patients and there is always a team of one certified nurse and two unlicensed assistants in each of the two sections.

The unlicensed assistants are not allowed to dose and report patient condition which means that Mary is responsible in her team to handle the reporting and diagnosis and the only one who regularly need to use the computer. Therefore she views the laptop in her section of the ward as *hers*. She uses the ward's public passwords when possible and for other applications she uses the password "p" (so she can log in using only one hand).

Mary [55]

Scania University Hospital often have trainees and since Mary has been there for 3 years she will often supervise one of them. When she supervise she tend usually first show the trainees how they should work and then give a couple of shortcuts on how to make it faster. Although she will stress that these shortcuts should only be used when they lack time to use the right way.

A.2.1 Mary's Goals

- I want to give patients the care they need and
- report patient status to my colleagues so that we together
- avoid misunderstandings and mistakes.

A.2.2 What Do Mary Want From Us

Start working immediately Mary's work can be pretty stressful and sometimes she will not have time to sit down and document until quite a while later. It is important to get into the system immediately and without distractions so she don't forget details.

Save sessions Mary often have to leave abruptly to tend to emergencies. When returning she wants to resume where she left off.

Help follow policies Mary want to follow the rules and regulations that exists. But when her workload rises she always takes the easiest way, priority is on patients. Therefore she doesn't want to worry about working the wrong way while she is focusing on her patients.

A.3 Billy

Billy works as a salesman in a major retail store with storehouses all over Europe. He started working five years ago in the electronics department but has since moved to kitchen appliances.

Billy [56]

There's usually three salesmen in kitchen appliances and they share the two computers used to order wares and assist customers. Billy is the most senior among all salesmen in his department so when the other salesmen need help with assisting a customer they often come to him. Billy knows most products in their inventory and he has learned the best shortcuts in the computer interface.

Billy is a *people's person* and he likes talking to and assisting customers, even when it turns out they need something he is not able to provide. He believes it is better to not sell something they customers do not need and building a customer trust in their brand name.

A.3.1 Billy's Goals

- I want to make sure the customers get what they need, and
- making as much profit as possible, all the while
- maintaining an effective working structure.

A.3.2 What Do Billy Want From Us

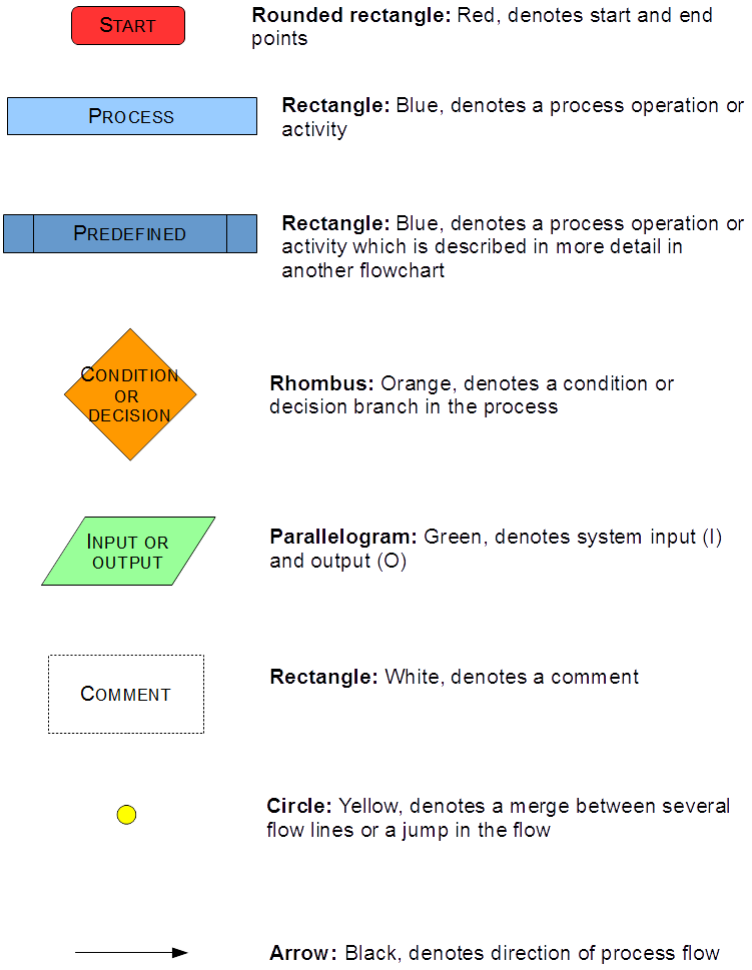
Getting started immediately There is always a steady stream of customers, He often takes them to the aisles to show them their products. This means he will arrive at the workstation often. Some days more than a hundred times. Therefore it is important for Billy to get started and booted up instantly so customers don't have to wait and build up a queue.

Save sessions When leaving the computer Billy can't leave the computer logged in for two reasons. First customers shouldn't have access to their internal system. and secondly his coworkers might need it will he is gone and they should authenticate themselves for audit purposes. But Billy don't want to restart from the beginning every time he returns. Therefore he want to be able to save or freeze sessions.

Not bound to one computer Since Billy and his coworkers share two computers he want to be able to migrate his work from one computer to the other if he returns and one of his coworker is busy with the computer he originally worked on.

Appendix B

Flowchart - Legend



Flowchart Legend

LUND
UNIVERSITY

Series of Master's theses
Department of Electrical and Information Technology
LU/LTH-EIT 2015-457

http://www.eit.lth.se