Designing Usable and Secure Authentication Mechanisms for Public Spaces

Dissertation

an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig-Maximilians-Universität München

vorgelegt von
Diplom-Medieninformatiker
Alexander De Luca

München, den 15. März 2011

Erstgutachter:	Prof. Dr. Heinrich Hußmann
Zweitgutachter:	Prof. Dr. Marc Langheinrich

Tag der mündlichen Prüfung: 30. Mai 2011

ABSTRACT

Usable and secure authentication is a research field that approaches different challenges related to authentication, including security, from a human-computer interaction perspective. That is, work in this field tries to overcome security, memorability and performance problems that are related to the interaction with an authentication mechanism. More and more services that require authentication, like ticket vending machines or automated teller machines (ATMs), take place in a public setting, in which security threats are more inherent than in other settings. In this work, we approach the problem of usable and secure authentication for public spaces.

The key result of the work reported here is a set of well-founded criteria for the systematic evaluation of authentication mechanisms. These criteria are justified by two different types of investigation, which are on the one hand prototypical examples of authentication mechanisms with improved usability and security, and on the other hand empirical studies of security-related behavior in public spaces. So this work can be structured in three steps:

Firstly, we present five authentication mechanisms that were designed to overcome the main weaknesses of related work which we identified using a newly created categorization of authentication mechanisms for public spaces. The systems were evaluated in detail and showed encouraging results for future use. This and the negative sides and problems that we encountered with these systems helped us to gain diverse insights on the design and evaluation process of such systems in general. It showed that the development process of authentication mechanisms for public spaces needs to be improved to create better results. Along with this, it provided insights on why related work is difficult to compare to each other. Keeping this in mind, first criteria were identified that can fill these holes and improve design and evaluation of authentication mechanisms, with a focus on the public setting.

Furthermore, a series of work was performed to gain insights on factors influencing the quality of authentication mechanisms and to define a catalog of criteria that can be used to support creating such systems. It includes a long-term study of different PIN-entry systems as well as two field studies and field interviews on real world ATM-use. With this, we could refine the previous criteria and define additional criteria, many of them related to human factors. For instance, we showed that social issues, like trust, can highly affect the security of an authentication mechanism.

We used these results to define a catalog of seven criteria. Besides their definition, we provide information on how applying them influences the design, implementation and evaluation of a the development process, and more specifically, how adherence improves authentication in general. A comparison of two authentication mechanisms for public spaces shows that a system that fulfills the criteria outperforms a system with less compliance. We could also show that compliance not only improves the authentication mechanisms themselves, it also allows for detailed comparisons between different systems.

ZUSAMMENFASSUNG

Passwort und PIN (personal identification number) sind der heutige de facto Standard zur Authentifizierung. Im Forschungsfeld der “ usable and secure authentication” werden verschiedene Herausforderungen und Probleme von Authentifizierung aus der Sicht der Mensch-Maschine-Interaktion angegangen. Das bedeutet, es wird zum Beispiel versucht, Sicherheits-, Erinnerbarkeits- und Leistungs-Probleme zu beheben, die aus der direkten Interaktion eines Menschen mit der Authentifizierungstechnik entstehen. Mit der steten Zunahme von Diensten, für die es nötig ist sich zu authentifizieren, wird dieses Problem ständig größer. Zusätzlich spielen sich immer mehr dieser Dienste in öffentlichen Räumen ab, in denen vor allem Sicherheitsprobleme akuter sind als an privaten Orten. Zu diesen Diensten gehören vor allem Fahrkarten- und Geldautomaten. In dieser Arbeit haben wir uns deswegen entschieden das Problem der “usable and secure authentication” mit einem Schwerpunkt auf öffentliche Räume anzugehen.

Das Hauptergebnis dieses Arbeit ist ein Set von Kriterien für die systematische Evaluierung von Authentifizierungsverfahren. Diese Kriterien basieren auf zwei unterschiedlichen Ansätzen von Untersuchungen. Zum einen wurden prototypische Implementierungen von Authentifizierungsverfahren entwickelt, welche die Sicherheit und Benutzbarkeit heutiger Systeme verbessern. Auf der anderen Seite haben wir empirische Studien zu Authentifizierung in öffentlichen Räumen durchgeführt. Dementsprechend liegt dieser Arbeit folgende Strukturierung zu Grunde:

Wir haben fünf verschiedene Authentifizierungsverfahren entwickelt, um die Schwachpunkte und Probleme verwandter Arbeiten zu beheben basierend auf einer neuen Kategorisierung von Authentifizierungsverfahren für öffentliche Plätze, welche für diese Arbeit entwickelt wurde. Die Prototypen wurden bis ins Detail evaluiert und lieferten gute Ergebnisse. Auch die negativen Aspekte dieser Arbeiten waren von Nutzen. So konnten wir Einblicke in den Design- und Evaluierungs-Prozess solcher Systeme gewinnen. Diese zeigten, dass der Entwicklungsprozess weiter verbessert werden muss, um bessere Ergebnisse zu liefern. Zusätzlich zeigte es sich, warum verwandte Arbeiten schwer vergleichbar sind. Neben diesen Erkenntnissen konnten wir erste Kriterien identifizieren, welche das Potenzial haben den Entwicklungsprozess von Authentifizierungsverfahren zu verbessern, vor allem mit Fokus auf öffentliche Plätze.

Weitere Einblicke, vor allem mit Hinblick auf Faktoren, welche die Qualität eines Authentifizierungsverfahrens beeinflussen, konnten wir anhand einer Reihe weiterer Arbeiten gewinnen. Diese bestand aus einer Langzeitstudie über die Nutzung verschiedener PIN Verfahren sowie zwei Feldstudien und Interviews über Geldautomatennutzung. Basierend auf diesen Studien konnten wir die Kriterien des ersten teils dieser Arbeit verifizieren. Außerdem wurden weitere Kriterien definiert, von denen viele menschliche Faktoren repräsentieren. Ein Beispiel sind soziale Bedingungen, wie Vertrauen, welche einen großen Einfluss auf die Sicherheit von Authentifizierungsverfahren haben können.

Basierend auf diesen Ergebnissen präsentieren wir einen Katalog von sieben Kriterien. Neben einer exakten Definition dieser Kriterien findet sich darin auch eine Anleitung, wie deren Anwendung den Entwurf, die Implementierung und die Evaluierung während des Entwicklungsprozesses beeinflussen oder besser gesagt, wie deren Einhaltung Authentifizierungssysteme verbessern kann. Mit einem Vergleich von zwei Authentifizierungsverfahren haben wir gezeigt, dass ein System, dass die Kriterien erfüllt ein anderes System in Sicherheit und Benutzbarkeit übertrifft. Außerdem konnten wir zeigen, dass die Befolgung der Kriterien die Systeme nicht nur verbessert sondern es auch erlaubt diese einfacher miteinander zu vergleichen.

ACKNOWLEDGMENTS

The work presented in this thesis covers a period of approximately four years. It would have been literally impossible to handle all the workload doing it alone. Fortunately, I was supported by several great people, which can be found listed as co-authors of the respective papers. Therefore, I decided to use the scientific plural in this thesis, which I consider more adequate in this context.

During the four years, a lot of amazing people crossed my path and many supported me in various ways, some of them not directly connected to my work. Firstly, I want to thank my two supervisors. Professor Dr. **Heinrich Hußmann**, my supervisor and boss, is one of the brightest and most patient people I ever worked with. I want to especially thank him for his excellent feedback, which he provided whenever I needed it. Professor Dr. **Marc Langheinrich** was my second and external supervisor. I want to thank him for all his valuable feedback in all stages of my thesis. I was always impressed by how nicely and easily he could express even the most complex aspects of my work and the effort he spent on co-authoring our joint paper.

Of course, I want to thank my current and former colleagues, many of which became good friends during these years. Special thanks go to **Albrecht Schmidt** and **Enrico Rukzio** (for introducing me to science and giving me my first insights on the DOs and DON'Ts of our field), **Otmar Hilliges** (for sharing my passion for FC Bayern München and for being a great friend), **Richard Atterer**, the good soul at Google (for being a brilliant person, a great friend and for being my best man), **Andreas Pleuß** (for many distracting and interesting conversations), **Heiko Drewes** (for heated discussions that often turned out very insightful), **Paul Holleis** (for teaching me how to write a good CHI rebuttal), **Sebastian Boring** aka Schlandmann (for sharing his football expertise with me), **Sara Streng** (for being the best office mate ever), **Gregor Broll** (for being a willing conversation partner when it comes to complaining but mainly for his passion for tennis, which helped me getting to know my wife), **Dominikus Baur** (for singing and fighting with me) and **Max Maurer** (for sharing my research interests and for the hot pot).

During those years, I also got to know great people from different countries, many of which I worked with. Amongst them, I want to mention **Petteri Nurmi** (for being a great friend, endless hours of conversations and for hosting the best volcano party ever), **Paul Dunphy** (for his hospitality and using MoodyBoard in his lectures), **Anja Thieme** (for being the funniest party guest ever and for letting us work on her jigsaw puzzle), **Roel Peters** (the coolest Belgium guy I ever met), **Tim van Kasteren** (for lots of fun basically everywhere in the world, I hope once we will meet in Germany), **Julian Seifert** (for being the best student volunteer ever and for his awesome work on TreasurePhone), **Marko Jurmu** (mostly for being a co-inventor of the network dance), **Roman Weiss** and **Bernhard Frauendienst** (the most productive students ever).

I also want to thank my old friends and my family for being with me all these years. Especially **Thomas Lang** and **Roberst Stöckle** (for always being there for me and being awesome friends since forever), my grandmother **Luise Lieb** (I hope you are proud of me and always keep an eye on me from up there), my aunt **Anneliese Rauner** (for treating me like a second mother), my father **Ezio De Luca** (for being a good friend and always having an open ear for my problems),

and my mother **Erika De Luca** and my stepfather **Klaus Wahle-Eggers** (for always supporting me and spending a fortune on my education).

Last but most definitely not least, I want to thank my wife **Xueli An-De Luca**, the center of my life and my greatest support. I cannot imagine anyone more patient. You not only accepted endless nights of working on papers and on my thesis, you also tried to help me wherever you could. You getting the first PhD in our family made me even work harder on my thesis. I am very proud of you and if you do not deserve a summa cum laude, then no one does. I love you.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

Introduction

The bank puts up cameras, so I am safe!

– an anonymous user study participant –

Since the first automated teller machine (ATM) was installed in the late 60s, public terminals provide a convenience that most users do not want to miss anymore. Their main benefit is to provide services 24 hours, seven days a week, which makes them become more and more part of our everday life. Nowadays, public terminals cover uncountable scenarios including ticket vending machines, quick check-in kiosks and self-service gas stations.

Whenever it comes to sensitive services like withdrawing money from an ATM or buying train tickets with a credit card, users have to authenticate themselves to the terminal to prove that they have the rights to perform the desired action. The most common approach for this is a so-called two-factor authentication. In the case of an ATM or a ticket vending machine, this means using a bank card, like a credit card as “something you have” plus a corresponding personal identification number (PIN) as “something you know”. Depending on the service, the authentication token can vary significantly and create a burden for the users in case they have to remember a larger number of them. This can easily lead to user caused security problems [2], for instance, by writing down the authentication token, or by using the same authentication token, like the same PIN, across different services or using words that can be found in dictionaries. An impressive example of this was shown by Klein, who could crack 25% of 14,000 passwords using a dictionary attack with only 86,000 words [76]. This highlights the problem and thus, it is not surprising that the user is often referred to as the ’weakest link’ in the security chain [111].

Despites the problems with authentication tokens, the manifold kinds of services that are available at public terminals and the resulting variety of scenarios, there is one predominant attribute that they all have in common. They are located in the same setting, a *public space* with all its problems, such as manipulations of the terminal.

1.1 Public Spaces

The most common approach to categorize public spaces is to differentiate between public, semi-public and private spaces [15]. Public spaces (as opposed to private spaces) are defined as places that are accessible to everyone without any special limitations. Semi-public spaces are similar to public spaces with the difference that stricter rules apply to them. For instance, a sports stadium can be considered a semi-public space, since an entrance fee has to be paid to use the space. This categorization is also often based on the users' feeling of control and possession [81]. Interestingly, people tend to use different "faces" or "masks" depending on their current context. Such a face defines what information a person reveals (or is willing to reveal) to a specific audience or in a specific setting [55].

In this work, however, public spaces are defined much more straightforwardly. A public space is an area, in which a specific service takes place and which is accessible by an arbitrary (not closely defined) number of people without any major controlling instance limiting access to the respective space. This means that within this definition, a bank would count as a public space while a user's home would be a private space. A location that would usually be defined as a semi-private space, like the previous mentioned sports stadium, counts as a public space in the context of this thesis. The two main attributes of a public space are therefore *simple access* and *presence of other persons* as a potential threat to the user.

At first glance, a binary definition of the term public space as used in this work might look like an extreme simplification that does not cope with the complexity of users and their perception of privacy, security and personal space. Within the scope of this work, however, it is sufficient. The most important attribute of a public space as defined here is its openness to a huge number of potentially (and most likely) unknown persons that might be a threat to the users' security, privacy or both. That is, we define a public space as a security problem that has to be considered and overcome when designing authentication mechanisms for this specific setting.

1.2 Problem Statement

The dominating approach to authentication nowadays is the use of secret passwords or PIN (personal identification number). Security problems of these approaches are the main driving force of scientific effort in the area of authentication mechanisms. For instance, the concept of using a four-digit PIN to authenticate is very easy to understand and allows for very fast interaction. At the same time, however, it is very insecure since there is no protection mechanism built-into the authentication process (besides showing asterisks on the screen instead of the real input). That is, the "secret token" is input in plain view of anyone around.

As just discussed, the main problem is the fact that the authentication process takes place in a public setting and can thus easily be spied on or be manipulated. This complex of problems is emphasized by the huge number of reported ATM frauds every year. Due to their property of allowing direct access to a bank account and thus to the users' savings, ATMs are the number one target for criminal activities and thus will be the most cited example within the scope of this thesis. In 2009, there were 2,058 reported manipulations of ATMs in Germany which led to more than 100,000 users becoming victim to ATM frauds. This is a plus of around 20% compared to the previous year. This trend is continuing in 2010[1].

These statistics only include attacks that were due to direct manipulations of the ATM, so-called "skimming" attacks. They rely on physical manipulations of the terminal by, for instance, adding fake keypads to record the PIN, and readers to copy the magnetic information of the card. Another often used skimming method is adding hidden cameras to the ATM to record the input of the users. Shoulder surfing, a very common attack [104], is not even covered by these statistics. A shoulder surfer tries to spy on the user's input from a close position. Again, the public space is what makes such a simple attack possible and so effective.

A different problem that typically comes with the use of standard PIN or password is the users themselves being a threat to the system and compromising its security. In the field of usable privacy and security, a sub-field of human-computer interaction (HCI), the common believe is that teaching the users to behave more secure, e.g. by using punishment, will not solve this problem as already stated by Adams and Sasse [2]. The best solution seems to be considering the users when designing authentication mechanisms ("user-centered design"). Thus, in this work, properties related to behavioral factors are outlined as important criteria that have to be considered when designing such systems.

As a result of the weaknesses of PIN and password in their current form, research in this area often focuses on how to make authentication mechanisms more secure. Evaluations of these new authentication systems are usually based on standard usability criteria like speed and error rate. Security is evaluated on different levels. It is possible to find purely theoretical security analyses considering factors like guessability and password space. Additionally, evaluations sometimes consider and simulate shoulder-surfing and skimming attacks as well. Due to its complexity, memorability evaluation is seldom found even though, as will be shown later in this work, memorability is an important aspect that does not only influence usability of a system but can significantly influence security as well [33]. This shows that within the field of usable and secure authentication, there is still a lack of understanding on how a perfect authentication system has to look like and more importantly, how and based on which properties it should be judged and evaluated. Lacking this understanding, it is very hard (or impossible) to compare related work to each other since the focus as well as the chosen evaluation approach differ significantly.

Summarized, there are two major problems with authentication for public spaces that we try to solve with this thesis. Firstly, it is insecure and there is still lack of appropriate and usable solutions to replace standard PIN or password. Second, evaluating authentication mechanisms is not an easy tasks and current approaches significantly differ from each other which makes comparing different systems rather hard.

[1] Press release of the German Federal Criminal Agency (May 25, 2010)
http://www.bka.de/pressemitteilungen/2010/pm100526.html

1.3 Authentication Scenarios

Authentication in public spaces is not limited to ATMs. Another important aspect of public terminals and services is therefore the huge variety of possible scenarios for which authentication can be required. Withdrawing money from an ATM is one of the most important interactions with public terminals nowadays and thus will be the major example throughout this thesis. Nevertheless, there are other services that are commonly used and already widely deployed and the future promises many more. Especially in the area of ubiquitous and pervasive computing [131], new kinds of public services that are, for instance, involving context data of a user, are emerging or proposed by researchers.

Amongst the most known examples of current public services are ticket vending machines of all kinds. Especially in the sector of public transport, they are becoming the main source for retrieving tickets. This has already reached a point in which in some areas, ticket kiosks are completely replaced by terminals. Authentication becomes necessary whenever users want to pay a ticket by electronic means (like a credit card) instead of cash. In some instances, ticket vending machines do not accept cash at all anymore or offer discounts to the users in case they use electronic payment. Another example are automated package stations that enable users to receive a delivery after authenticating to the terminal. Using a ticket vending machine or a package station is significantly different from using an ATM. Still, they use the same authentication mechanism.

Considering recent advances in ubiquitous technology, the next generation of public terminals and services will be much more versatile than what we currently see. Users will be able to connect to services, e.g. using their mobile devices, and adapt them to their needs and context. Adaptive navigation systems using public screens [77, 108] and games on public displays [120, 129] are just some examples that will most definitely require authentication of some sort once they grow up, leave the lab and hit the market. In these examples, mobile devices will very likely play an important role which raises the question how to involve them into the authentication process.

This variety has several implications on the design of authentication mechanisms for public spaces. For instance, when a system is proposed, it should be stated for what scenario it has been envisioned. If it involves mobile devices, connection establishment has to be considered which influences both, security and usability. Another important influence of the scenario is availability of hardware at the terminal and whether it is feasible to extend the hardware for the needs of the proposed system.

1.4 Main Contributions

This thesis mainly contributes to the field of authentication mechanisms for public spaces in two ways: It improves usable and secure authentication mechanisms and describes an extended development process based on criteria that influence design, implementation and evaluation of the respective systems.

1.4.1 Authentication Mechanisms

This work gives an overview of current approaches on secure authentication mechanisms and presents an analysis and categorization of existing related work. Furthermore, within the scope of this work, five secure authentication mechanisms have been created. They are explained in detail from design stage to evaluation containing a description on why the specific approaches have been chosen and how they are appropriate for specific scenarios (or not). Additionally, those examples are used to highlight problems that come with designing secure and usable authentication mechanisms for public spaces as they have been encountered first hand. Working on the design and evaluation of these prototypes helped to identify weaknesses in the state-of-the-art evaluation process of authentication mechanisms.

Additionally, the systems fall into the three main categories of authentication mechanisms for public spaces as discussed in chapter 2: software, hardware and user owned device based approaches. Each of the authentication mechanism solves specific problems that come with the respective category and present potential solutions within the scope of the different categories.

1.4.2 Design and Evaluation

The thesis also provides a detailed analysis of what criteria are important when it comes to the design and evaluation of authentication mechanisms, with a focus on public settings. Those criteria are based on the analysis of related work and evaluation techniques and practices that are currently employed for usable authentication as well as their problems. Existing criteria like security, memorability and usability in general are discussed and extended to fill gaps that were identified during this work. Further criteria were gathered by the design and evaluation of five secure authentication mechanisms, performing a long-term online study on PIN usage and conducting a series of field studies on real world ATM use. The so-defined criteria extend existing approaches. Additionally, new criteria are defined that mainly focus around behavioral aspects of authentication that come with the use of authentication in public spaces. This way, this thesis tries to move the creation of authentication systems closer to human-centered design.

The criteria have been applied in practice which outlines how they provide the following benefits:

1. They can significantly help to improve, judge, reject or accept designs throughout the whole development process. More importantly, this can already be achieved in a very early stage.
2. They provide a checklist to improve evaluation of authentication mechanisms and thus make them comparable to each other.

Finally, it should be highlighted that the here presented criteria strictly focus on aspects of usable and secure authentication relevant for human-computer interaction. Therefore, psychological and design-oriented issues are the main focus, whereas many lower-layer technical aspects of security, like encryption, will not be covered by the criteria.

1.5 Structure

This thesis is structured into three main parts, “authentication mechanisms” (chapter 2 and 3), “field studies” (chapter 4 and 5) and “criteria and use cases” (chapter 6). As mentioned before, the two main contributions of this work are newly proposed secure authentication mechanisms that solve problems of existing solutions and a new development approach based on seven criteria.

Contribution one is mainly presented in the first main part, which categorizes and discusses problems of related work (chapter 2) and proposes possible solutions that have been developed in the scope of this work (chapter 3). Part two and three cover the biggest part of the second contribution – improving the development process of authentication mechanisms – by presenting two major field studies (chapter 4 and 5) that helped to close gaps that we identified during the evaluation of our authentication mechanisms. Finally, based on the preceding parts (including the work on authentication mechanisms, which helped to identify first criteria), the criteria are defined and supported by two exemplary use cases (chapter 6).

In detail, the chapters present the following:

> **Chapter 2:** This chapter gives an overview of existing work in the field of authentication mechanisms. We describe and categorize secure authentication mechanisms including the different advantages and drawbacks inherent to the specific categories. Besides that, this chapter has several additional functions. It highlights how authentication mechanisms are currently evaluated and how an improvement of these approaches can lead to more insights and better comparable results. Additionally, it outlines benefits of biometrics but at the same time discusses major issues that still disqualify it for major applications (besides its many benefits). The field of graphical authentication mechanisms is introduced. This is important since they present a huge field of research focusing on improving memorability. Additionally, this definition is required to separate this field to mechanisms designed to improve security. Finally, we describe work performed on the analysis and evaluation of behavioral factors and how they influence authentication, especially in public settings.
>
> **Chapter 3 – Authentication Mechanisms:** Based on the categorization of secure authentication mechanisms for public spaces, in chapter 3 we discuss five authentication mechanisms that we designed to overcome the weaknesses of current authentication approaches as identified in chapter 2. All systems were evaluated and proved to provide several benefits. The chapter closes with a discussion of first criteria, remaining problems and most specifically remaining gaps that require additional work to be resolved.
>
> **Chapter 4 – Evaluating PIN and its Influence on Standard Usability Factors:** To get a grip on basic factors and problems of evaluating authentication mechanisms, we conducted a long-term usability study of different PIN-entry system, which is presented in chapter 4. Studying such a basic and well-known system allowed for precisely evaluating every single detail which in turn was very helpful for defining important criteria. One of the outcomes is a time measurement approach based on several phases that can

reveal important findings and in turn lead to partially rejecting randomization as an appropriate tool to make authentication mechanisms secure.

Chapter 5 – Authentication in the Wild: In this chapter, we describe a long-term field study on ATM interaction together with two follow-up studies, public interviews and a second field study. Those studies are based on the assumption that they can reveal important human-related factors rather than technological factors on authentication in public spaces. In the outcome of the chapter, we describe were several findings like the absence of secure behavior and factors that have major impact on them. Based on these, implications have been derived that have directly influenced the creation of human-centered criteria that we propose within this thesis.

Chapter 6 – Criteria and Use Cases: The practical work performed during this thesis resulted in a set of criteria, which we describe in this chapter. Firstly, the criteria are derived from the results of the previous chapters. Following this, they are located withing a standard development process of an authentication mechanism for public spaces. We exemplarily show the benefits of the criteria by applying them to two authentication mechanisms created in this work.

Chapter 7: We conclude with a detailed summary of the main contributions of this thesis on both, improving secure and usable authentication for public spaces and providing an improved development and evaluation process for authentication mechanisms based on extended and new criteria. Following this, we discuss how this work has closed existing gaps and at the same time opened new one, thus providing new research questions for the future. In the end, the influence on this work on other areas of usable privacy and security is highlighted and exemplarily accentuated by discussing additional future work besides authentication mechanisms.

Chapter 2

Related Work:

Authentication Mechanisms and their Evaluation

Learn from the mistakes of others. You can't live long enough to make them all yourself.

– Eleanor Roosevelt –

This quote by Eleanor Roosevelt expresses the standard approach of good science. Instead of reinventing the wheel, we should be standing on the shoulders of giants. Therefore, this chapter summarizes important work done in the field of usable and secure authentication and defines how the work performed in this thesis relates to it and builds upon it. It also highlights how this thesis can fill gaps that have not been filled yet. Finally, it discusses first approaches that have been conducted to integrate behavioral aspects in the design of authentication systems and closes with how these findings helped to bring research further.

2.1 Biometrics, not the Holy Grail yet

Biometrics [4] is often considered the holy grail of authentication mechanisms. It has the potential to easily solve several of the main security and usability issues of authentication. For instance, having a physical property of a user as the authentication token, it is literally impossible to forget it. Furthermore, biometrics are usually fast and intuitive. With the technology getting cheaper and more reliable, the factor cost is constantly becoming a smaller issue as well.

Besides all the benefits, biometrics has three major problems that, till now, have prevented it from wide acceptance and deployment: privacy, legal and economical issues.

Privacy issues are related to manifold concerns regarding biometric authentication from a user perspective. All the studies performed in this area show that there is a common level of mistrust toward biometrics [23, 103]. Especially users that never used biometric authentication before state that they would not like to use those systems. Privacy concerns dominate the discussion since many users simply do not trust (or do not want to trust) a service provider to use their biometric features responsibly. In particular, the fact that this data cannot be changed and once it is lost, it will be lost forever, is a major concern [115]. Additionally, missing familiarity with the technology often raises health concerns when techniques like iris scans are discussed. Coventry et al. showed, however, that when users are exposed to biometric systems for a longer time, the acceptance of the system significantly increases [23]. Closely related to privacy issues are legal aspects that have to be taken into account. For instance, being an unchangeable token that cannot be revoked, the biometric feature has to be stored securely and the service provider has to make sure it will not be misused in any way. Another example is the fact that when collecting biometric data, information can be acquired that is unrelated to the authentication tasks and can give insights on the physical condition of the user. Thus, it has to be made sure that such information is deleted from the data set.

Economical issues arise from the fact that biometric information has to be recorded directly from the user and requires an infrastructure to maintain and distribute the information. That means that if, for instance, a bank wants to replace PIN with fingerprints, it has to collect them in person from all its customers which means that they have to show up personally at a branch. In a time where many banks do not even have physical branches anymore, this easily becomes a major economical factor. Updating thousands of terminals with new technology, even if the technology is cheap itself, is another economical issue that has to be taken into account.

Problems in deployment and the special user attitude toward biometrics are both reasons why they will not be thoroughly discussed in this thesis. Thus, fingerprint recognition, iris scans, novel approaches like hand-contour-based recognition [114] and the like will not be taken into consideration when talking about criteria and scenarios. Nevertheless, biometric authentication is a very interesting field, and though out of scope of this work, it surely deserves attention in the future.

2.2 Requirements for Authentication Mechanisms

There are three main requirements that are usually taken into account when designing and evaluating authentication mechanisms for public spaces: *memorability*, *security* and *usability* requirements. The standard approach to evaluate new authentication mechanisms is to compare them to PIN-entry (or sometimes password entry) in controlled laboratory experiments on the basis of these factors. Longitudinal experiments are also common, mostly to identify usability problems like memorability issues. Therefore, in the best case, the performance of these three factors should be better than for PIN-entry in evaluation.

Memorability

By definition, *memorability* is a usability feature. This requirements means that it should be easy for a user to remember how to authenticate. This mostly refers to the memorability of the authentication token. Due to the special role that it plays for authentication mechanisms, it is usually handled as its own category. Not only is memorability one of the most important usability factors, it additionally can directly affect the security of a system [2, 33]. For instance, when users have to remember complex authentication tokens, they tend to employ insecure behavior like writing them down or sharing them with others [2]. Therefore, memorability evaluation is considered a very important part of the evaluation process. Optimally, a memorability experiment should be conducted as a longitudinal study with multiple tokens [48]. However, due to the high complexity of realistically testing memorability and its huge demand on resources and time, memorability evaluation is often neglected and only theoretically approached.

Security

Security requirements usually take into account attacks as defined in a threat model. Threat models are the tool of choice to define against what attacks an authentication mechanism has been designed for. Thus, a theoretical security analysis is often used to “prove” resistance against dictionary attacks [102] and the like. A large password space is considered an important security criterion as well but is often smaller than mathematically calculated due to behavioral factors. For instance, when being allowed to define their own passwords or tokens, users only use a small portion of the available password space [125]. In other cases, more practical evaluations are employed based on different attacks. For instance, cameras are put up to simulate advanced skimming attacks [27] or shoulder surfing is employed to try an attack on the authentication mechanism [74]. Arbitrary combinations of these approaches can be found as well.

Usability

In addition to memorability, the *usability* factors error rate and authentication speed are typically analyzed. Errors are often categorized into different types. For instance, in the context of ATM authentication, basic and critical errors are distinguished [34]. Basic errors define one or two failed authentication attempts, while a critical error occurs when a user wrongly inputs the authentication token for three times in a row. The distinction comes from the fact that ATMs usually block the bank card as a safety precaution when three errors have occurred. In other cases, recoverable and unrecoverable errors [40] refer to errors that either just slow down authentication or make it completely impossible. Depending on the scenario for which the authentication mechanism has been developed, different types of errors are defined and evaluated.

For authentication speed, the situation is similar since different evaluation approaches can be found for it as well. There is a very old Italian saying. *“Ask 1000 Italians how to make a tomato sauce and you will get 999 different answers and answer number 1000 is from the brother of the person who gave the first answer.”*. For authentication papers that means: Read 1000 of them and you will find 999 different ways to measure authentication speed and paper number 1000

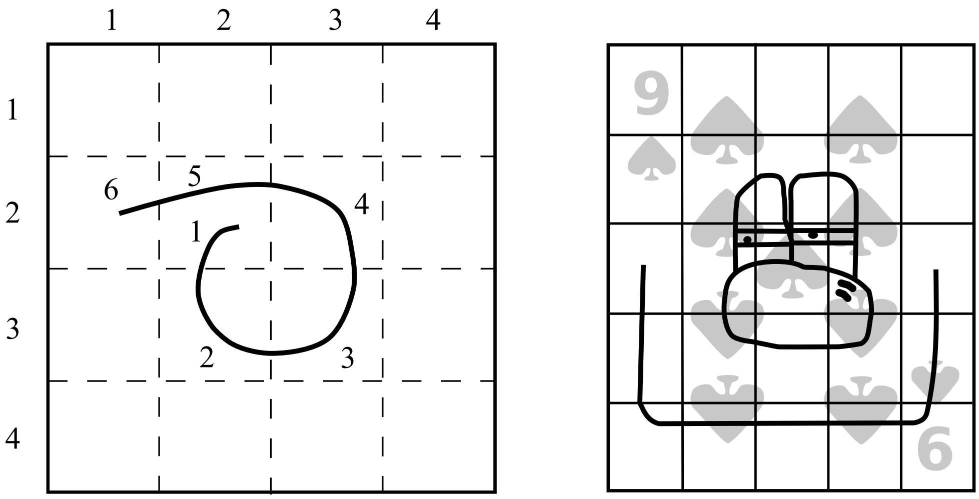


Figure 2.1: Left: An example of the original “Draw a Secret” authentication mechanism [67]. Right: An extended version using background images to improve several factors like memorability [45].

will most probably be a co-author of one of the other papers. Often, it is even impossible to find information on how exactly the measurements were done. The reason for that can be found in the fact that there is no agreement and only few information on what a “good” authentication speed for an authentication mechanism actually is and how it can be judged. This thesis will therefore give some deeper insights on the factors related to authentication speed that we could find and how incorrect (or imprecise) measurement can fail to reveal important usability issues of an authentication mechanism.

Problems of current Evaluation Approaches

Besides those requirements, only seldom other factors are evaluated. This is critical since this way, many behavioral factors that come with security problems are simply ignored. In this thesis, this gap will be further closed by taking a closer look at such factors and integrating them into the design process.

Another problem is the high diversity of authentication approaches. This makes it very hard to compare authentication systems to each other. Often, important information is missing in papers or has not been collected during evaluation. There is also missing consistency on how current factors are integrated into a study. For instance, security can be discussed theoretically only which makes it hard to judge the “real” security of the mechanism. To bring the solution of this problem one step further, within the scope of this work, the basic authentication criteria memorability, security and usability are defined and extended to highlight which factors should be evaluated and how. This can depend on the scenario and will thus be part of the criteria.

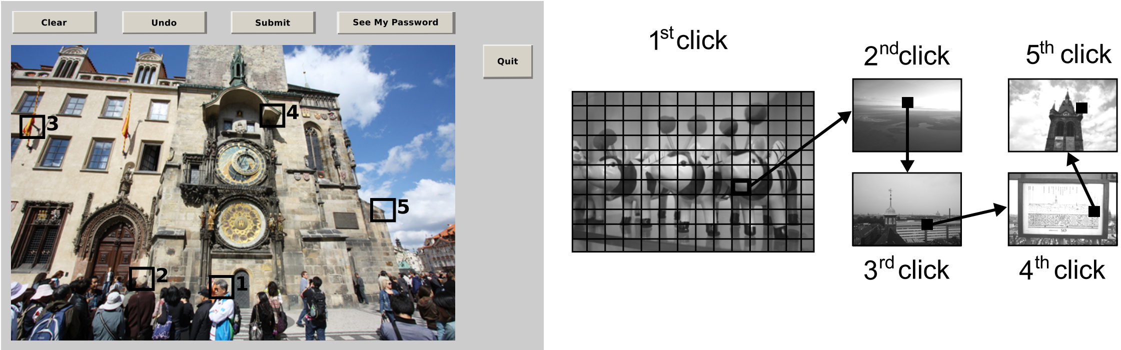


Figure 2.2: Left: An example of the PassPoints prototype [134]. To authenticate, users click specific points in the photo. Right: A typical Cued Click Points interaction scheme [17]. Authentication is done by selecting specific areas in a picture. Follow-up pictures are different depending on the selected area and thus give immediate feedback over success or failure.

2.3 Improving Memorability

Leaving biometrics aside, there have been manifold attempts to solve parts of the authentication problems as mentioned before. Two main directions of research can be distinguished. The first tries to improve authentication from a pure usability point of view, mostly focusing on memorability. The second focuses on solving security problems.

Most of the work on improved memorability can be found in the area of graphical authentication methods like picture based authentication mechanisms [26, 40] or drawing based graphical passwords [45, 67, 132]. Graphical authentication usually falls into one of the categories *drawmetric*, *locimetric* and *cognometric* systems [26].

Drawmetric systems are based on the users' ability to reproduce a predefined drawing. Especially repeated drawing of the same shape [45, 132] can significantly improve memorability employing the users' muscle memory [47, 117]. The best known system is "Draw a Secret" by Jermyn et al. [67]. An example is depicted in figure 2.1, left. To authenticate to the system, the user draws the authentication token on a 4x4 matrix. It is not necessary to reproduce the exact shape but how the shape is spread across the different areas of the matrix. The original system has several security weaknesses especially based on what kind of shapes users would choose and how this increases guessability of the tokens [97]. Dunphy et al. [45] could prove that background images can be used to improve memorability of the system as well as to get users to specify more complex shapes (e.g. figure 2.1, right), which was identified as a weakness of the original system.

Locimetric systems employ the loci-method [60] which uses spatial relationships to remember objects. These are also called *cued recall* techniques. A famous example is PassPoints by Wiedenbeck et al. [134] in which a user has to identify a predefined number of points in a picture as shown in figure 2.2, left. PassPoints was originally attested improved security due to the fact that it is resistant to dictionary attacks. However, Dirik et al. [42] could prove that an automatic dictionary attack on PassPoints is very easy and proposed the use of more resistant background

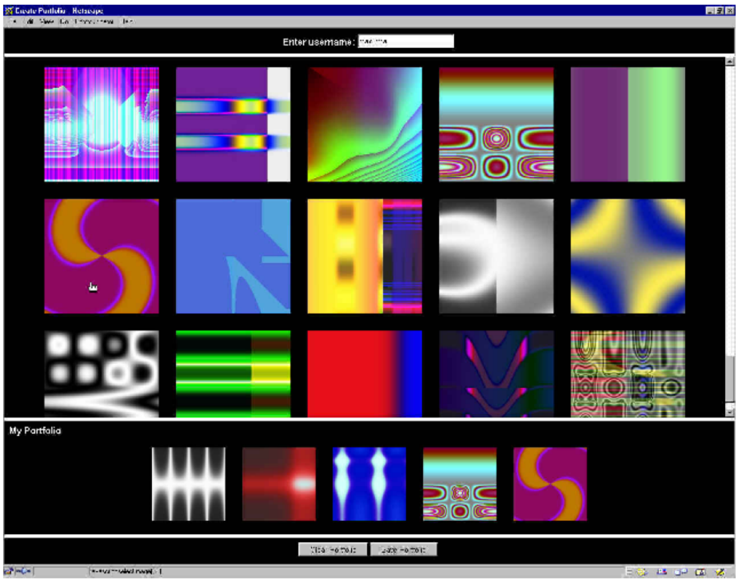


Figure 2.3: Random images in the portfolio creation of the Déjà Vu system [40].

pictures. Other work highlights how "hotspots", areas that attract high attention, can be used to attack the PassPoints system [126]. In addition to security, the choice of background pictures can also significantly improve the usability of the PassPoints system [16]. Interestingly, a study by LeBlanc et al. showed that eye gaze does not give insights on PassPoints choices [82]. In the Cued Click Points system by Chiasson et al. [17], the user has to recursively identify specific areas in given pictures (figure 2.2, right). Each area refers to a different follow-up picture which gives the users immediate feedback on whether they are doing the right input.

In the last category, cognometric systems, pictures are used as visual authentication tokens. This way these systems try to exploit the users' ability to easily recall something known [99]. Often, the pictorial superiority effect [98, 122] is used as an explanation why this works especially well for pictures. Examples include the commercial product Passfaces[1], VIP by De Angeli et al. [25] and Déjà Vu by Dhamija et al. [40]. In those systems, authentication is performed by selecting predefined images, photos and the like out of a bigger number of decoy images in several rounds. Figure 2.3 shows the portfolio creation screen of the Déjà Vu system. It uses random art instead of photos to minimize guessability of the authentication token, since for systems like Passfaces, it was found that users tend to choose photos of persons that they find attractive. This significantly decreases the password space and thus security [24]. Everitt et al. [48] could show impressive memorability results for photo based cognometric methods in a longitudinal experiment, even under hindered conditions with multiple passwords. An advantage of remembering multiple graphical passwords rather than multiple PINs was confirmed in a study by Moncur et al. [96]. Even though these systems already provide very good performance memorability-wise, researchers are still successfully working on improvements like the Composite Scene Authentication mechanism by Johnson et al. [68, 69, 70].

[1] Passfaces Corporation (last visited: August 11, 2010) http://passfaces.com/

The here discussed approaches mainly solve memorability issues and theoretical security issues like password space size. For this thesis, however, the focus on security aspects (theoretical and practical evaluations) and security/usability trade-offs when designing authentication mechanisms. Nevertheless, memorability is an important factor and has to be taken into account for the design of authentication mechanisms. Therefore, it will be an important part of the criteria discussed in this work. Moreover, memorability was often shown to directly compromise security [2, 33].

2.4 Improving Security

Even though memorability, as an important usability factor, deserves a lot of attention, security is certainly the most important feature of an authentication mechanism, especially for authentication in public spaces. Being the most critical problem of authentication, security aspects will also be the focus of this work.

The most widely used taxonomy of authentication mechanisms is to divide them into “something you have” (some material token), “something you are” (like a fingerprint) and “something you know” (a PIN or password). Arbitrary combinations of these categories can be found in deployed systems. For instance, coming back to our main example, the ATM, the standard authentication approach for this specific service is “something you have” (the bank card) plus “something you know” (the PIN). In some work, this taxonomy is extended by proposing new approaches like using “somebody you know”, for authentication [11, 113].

In the context of this work, we propose another categorization, which highlights the advantages and disadvantages of the different mechanisms in the context of public spaces. Additionally, this categorization is more accurate when it comes to assigning authentication mechanisms to specific scenarios. The three categories are *software based approaches*, *hardware based approaches* and *user owned device based approaches*.

2.4.1 Software Based Approaches

Software based approaches rely on clever software design to improve the users’ security during the authentication process. That is, no additional hardware of any kind is required. Usually, software based approaches use indirect input or other means of obfuscation to secure the input. Indirect input means that the authentication tokens are not directly input but instead some kind of “detour” is used. While most of the software based approaches are only able to secure the input against a shoulder surfer, since they are hard to spy on or being remembered when seen live, there are mechanisms that provide enhanced resistance, even to camera attacks.

The main problem of these systems usually lies in the big overhead that they add to the input or the high complexity that either significantly decreases input speed, increases error rate or both. However, they have the advantage that they do not require major changes at the terminal on which

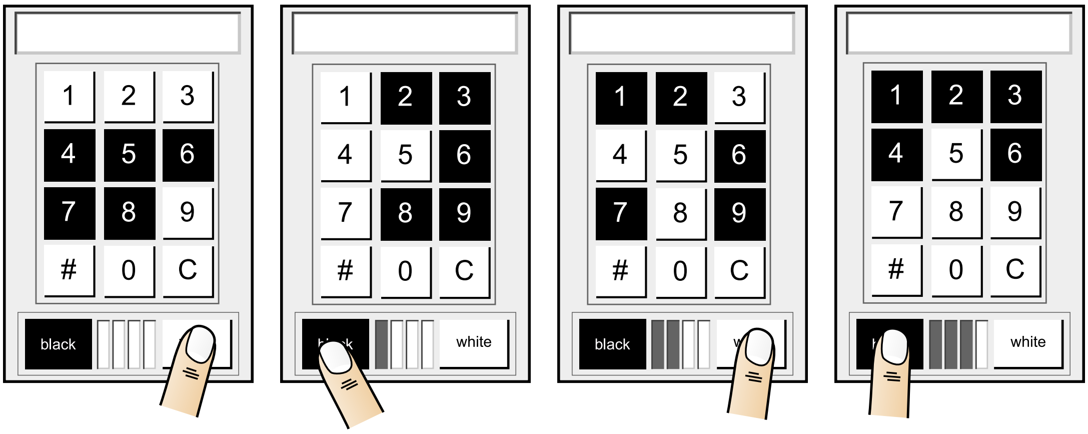


Figure 2.4: Four button presses are required to input a single digit of the PIN-entry method by Roth et al. [106]. In this example, the user inputs the digit ’3’. The figure is based on a series of screenshots depicted in [106].

they are supposed to run. Normally, standard software updates are sufficient. This is especially beneficial if a system has to be deployed to a large number of terminals in which case a major hardware update would considerably increase deployment costs.

Approaches designed to improve memorability are mainly based on graphical mechanisms, thus software based approaches (see chapter 2.3) dominate this field. However, designing a secure authentication mechanism based on pure software modifications is much harder and thus there are fewer approaches.

In 2004, Roth et al. [106] presented one of the earlier approaches in this field. Their system uses an obfuscation technique called “cognitive trapdoor game”. It uses multiple button presses to derive one digit out of a four digit PIN. For each press, the keypad is divided into five black and five white fields. The indirect input done by the user is pressing a button representing the corresponding color of the digit. To uniquely identify a digit, four rounds are required as shown in figure 2.4. That is, the input of a four digit PIN requires 16 button presses. This is a perfect example of how overhead, indirect input and visual distraction are used to confuse an attacker and thus make shoulder surfing nearly impossible. Even performed in plain view of an attacker, the authentication token stays protected. However, as with many software based authentication mechanisms, this system is not resistant to camera recordings. Thus, the authors propose extensions that could partially solve the problem. Due to the created overhead, this system performs significantly worse than standard PIN-entry in terms of input speed with around 25 to 35 seconds after a learning phase.

The “spy-resistant keyboard” by Tan et al. [124] uses a three step procedure based on obfuscation and indirect input to perform arbitrary input. Figure 2.5 shows the main phases of interaction. The most important property is that the final input is done while the characters are hidden (see figure 2.5 (right)). This way, the input is resistant to shoulder surfing attacks. Again, video attacks can easily reveal the user’s input. The system was evaluated for password input and compared

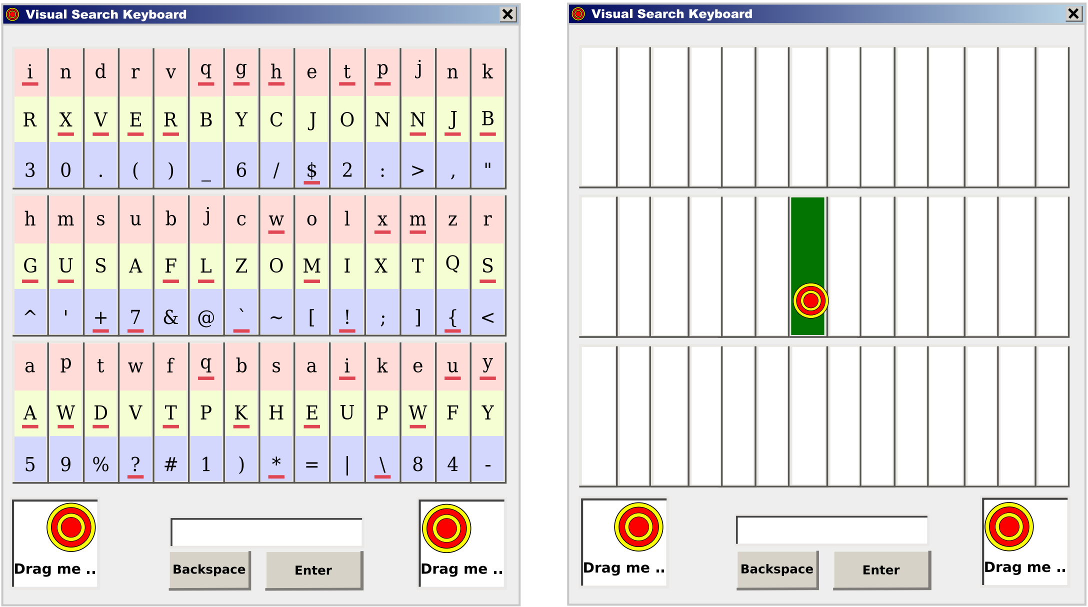


Figure 2.5: The spy-resistant keyboard [124]. On the left side, the standard screen is shown. Underlined letters are currently active. To change the underline state, the user presses a so called “Interactor”. Finally, the secret input mode is used to select the field with the correct character (right).

to a standard on-screen keyboard. The results show that with around 50 seconds input time, it is nearly two times slower than password entry on the on-screen keyboard. This can as well be attributed to the generated overhead.

In both cases, the cognitive trapdoor game as well as the spy-resistant keyboard, a previously discussed attribute of software based security enhancing authentication mechanisms is nicely highlighted. An overhead is added to the input effort to confuse an attacker and make the input resistant to shoulder surfing. This overhead however, negatively influences usability factors.

A system that performs slightly better, even over a longer period of time, has been presented by Hayashi et al. [59]. “Use your illusion” is based on cognometric systems like VIP [25]. The difference is that it uses visually distorted versions of the original images (see figure 2.6, right), which makes it highly resistant to shoulder surfing attacks. The authors could show that their system performs similarly to a clear image version in terms of memorability and error rate. Average authentication times in their study were between 12 and 26 seconds, which makes it a little faster than the previous discussed work. Unfortunately, the settings were too different to directly compare those times and make judgments based on this comparison.

The final authentication mechanism discussed in this section is the Convex Hull Click scheme by Wiedenbeck et al. [135]. What makes it different from the other mechanisms in this chapter is that it has been designed to be resistant not only to shoulder surfing but to camera attacks as well. This is achieved by applying an indirect input method. The user has to remember a set of pass-icons that are shown on the screen together with a group of decoy items as shown

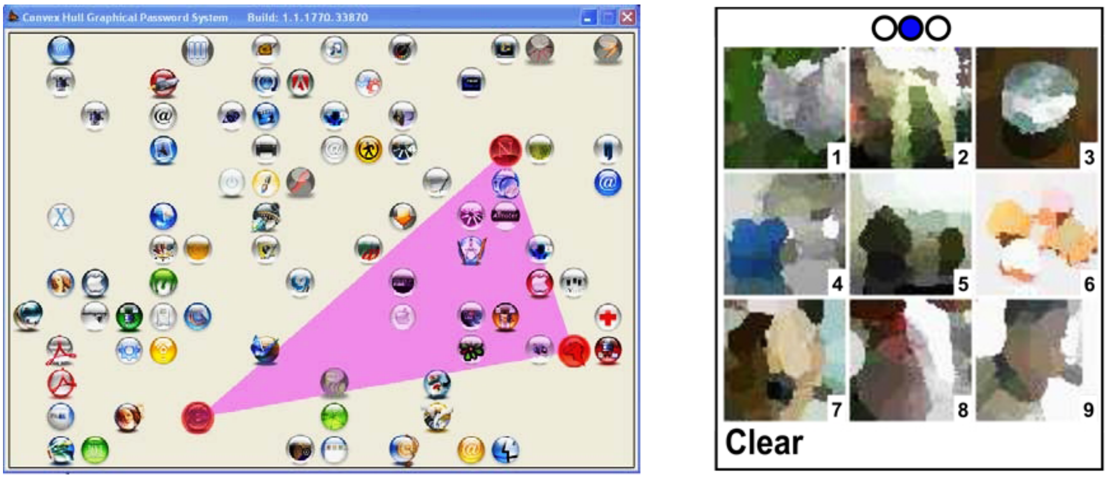


Figure 2.6: Left: An example of an input area of the Convex Hull Click scheme [135]. To respond to the challenge, users have to click within the area formed by their pass-icons (invisible during authentication). Right: Distorted images used in the "use your illusion" system [59].

in figure 2.6, left. To authenticate, the user has to mentally build a shape formed by the pass-icons and click in the invisible area defined by them. As many others, this is a typical challenge response authentication mechanism. However, the particularity of using invisible areas makes it highly secure. To break the system, several successive camera attacks are required to perform an intersection attack. Due to the increased mental effort of finding several pass-icons amongst the decoy icons, input time for a five round authentication in the simplest (and most insecure) configuration takes around 72 seconds on average. This means that again, the improved security comes with decreased performance.

Quickly summarized, it can be noted that software based authentication mechanisms take their special benefit from their low requirements on changes of the system they are installed on. Usually, a simple software update does the trick. Unfortunately, this benefit comes with heavily decreased performance compared to similarly secure mechanisms. It seems that the higher the security, the lower the performance. However, in this thesis, ColorPIN [32] will be presented (see chapter 3.4.1), a system that both improves performance compared to other software based approaches (especially in long-term use) and provides camera attack resistance similar to the Convex Hull Click scheme.

2.4.2 Hardware Based Approaches

In hardware based approaches, additional hardware is employed during the authentication process that helps to make a system resistant to manifold attacks. In most cases, this hardware is used to provide an invisible communication channel to or from the user to transfer secret information. Enhanced security is based on this information, which cannot or only with great effort be stolen

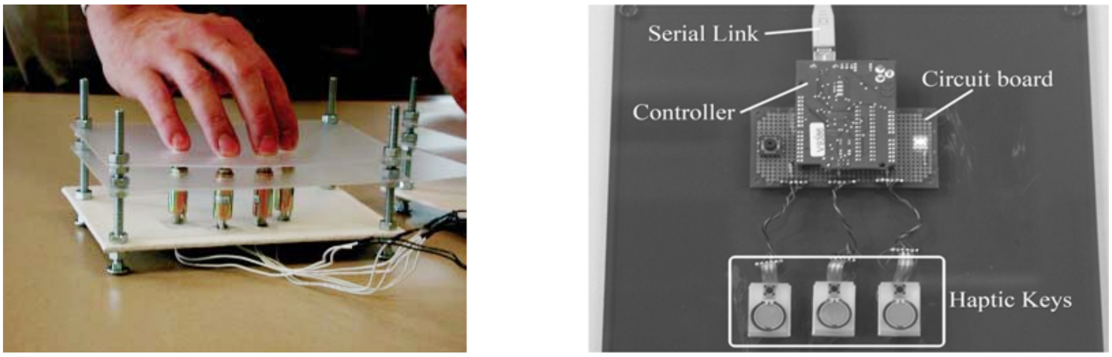


Figure 2.7: Left: The prototype of the tactile PIN-entry mechanism by Deyle et al. [39]. Passwords consist of a sequence of fingers. Right: A similar tactile system by Bianchi et al. [7], which uses a sequence of tactons (vibrotactile cues) as passwords.

by an attacker. The main advantage of these systems is that they have the potential to be extremely secure especially compared to software based systems. Performance-wise, the here presented approaches are comparable (or slightly better) than software based authentication mechanisms. Within a public context, however, these systems are extremely sensitive to manipulations of the hardware which causes extra security problems. Additionally, deployment costs are much higher than for software based systems. This is something that has to be kept in mind for their design and especially when considering appropriate scenarios for them. With other words: when designing for a specific scenario, these two drawbacks should always be kept in mind.

Delivering Secret Information to the User

The first examples in this chapter use hardware as an invisible communication channel to deliver secret information to the users. Based on this, the users can then infer the correct input that represents their password. In all those systems, the input, which can be spied on, is meaningless without the information that has been exchanged. Since this information is transmitted securely and invisibly, the systems are theoretically highly secure.

Hardware based authentication mechanisms that secretly transmit information to the user often rely on tactile information, that is, something the users can feel. In 2006, Deyle et al. [39] implemented and evaluated a system which uses a sequence of fingers as passwords. The users put their fingers over pins that are either lowered or raised (see figure 2.7, left). Authentication is done by selecting whether the current password-finger's pin is lowered or raised (using two buttons). To perfectly identify a finger, three rounds are required. Therefore, the system is similar to the cognitive trapdoor game by his co-author Volker Roth [106] in that it requires several rounds to identify a finger (digit) of the password. An attacker can only spy on the user input. Without the knowledge about the randomized position of the pins, this knowledge is useless. Unfortunately, no user study is published and thus, the performance of the system remains unclear. It can be expected, however, that the input speed is equally slow to the cognitive trapdoor game since the approaches are very similar. Security-wise it can be expected to perform much better.

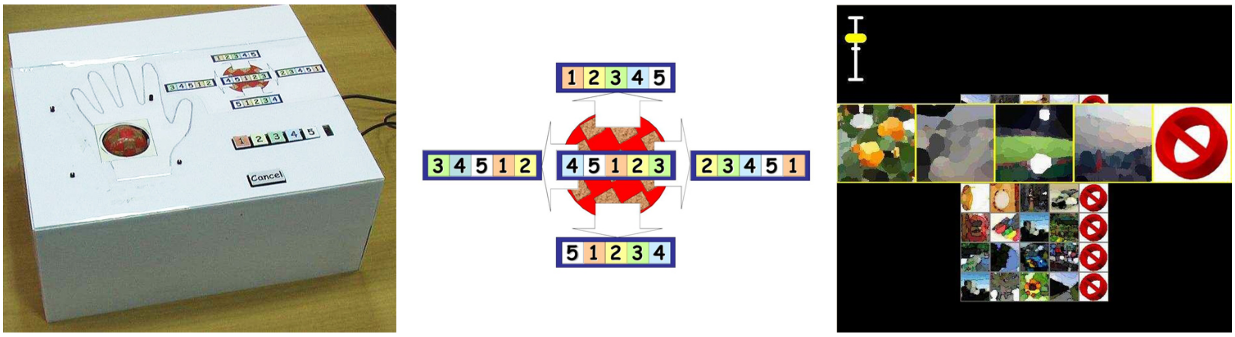


Figure 2.8: Left: The mechanical prototype of Undercover by Sasamoto et al. [110]. Center: The movement of the ball assigns a different order of five buttons to the images on the screen (right). This way, the secret information "ball movement" is mapped to a keypad layout.

Bianchi et al. [7] created a tactile authentication mechanism very similar to the just discussed approach, with the important difference that the password consists of a sequence of tactons [12] (vibration patterns that a user can distinguish). Three different tactons are randomly assigned to three keys as shown in figure 2.7, right. To authenticate, the users press the key that performs the current tacton. In a study with different password sizes, the system performed equal to the cognitive trapdoor game. It took on average 22 seconds for a 6-tacton password and 34 seconds for a 9-tacton password. The use of six and nine tactons instead of four as compared to a PIN, is needed to balance the small password space by only having three keys. Security can be rated high since the input does not give away any information on the users' password. The authors also spent large effort on other haptic based authentication mechanisms like the haptic wheel [8].

The final example of a system that uses secret information transmitted to the user has been developed by Sasamoto et al. in 2008. Undercover [110] uses a mechanical ball, hidden by the user's hand, to secretly transmit one of five keypad layouts as depicted in figure 2.8. The arrangement of the layout tells the users which button to press to select their pass image on the screen. This way, Undercover is a security enhanced version of a cognometric authentication mechanism as introduced in chapter 2.3. A study with seven challenges per authentication round revealed good security attributes but low performance with times between 32 and 45 seconds on average. However, the system is a good example of how a theoretically secure system can easily be compromised by its users. In this case, it can be argued that this is due to complexity since the authentication token of nine participants could be stolen due to reasons like not completely covering the ball or pointing on the respective keypad layout. That is, the participants opened security holes without noticing it.

Receiving Secret Information from the User

The second category of examples uses an invisible communication channel as well. The difference to the previous mentioned systems is that the channel is used to secretly transmit information from the user to the system. That is, in these authentication mechanisms, security is achieved by making the input of the password itself invisible.

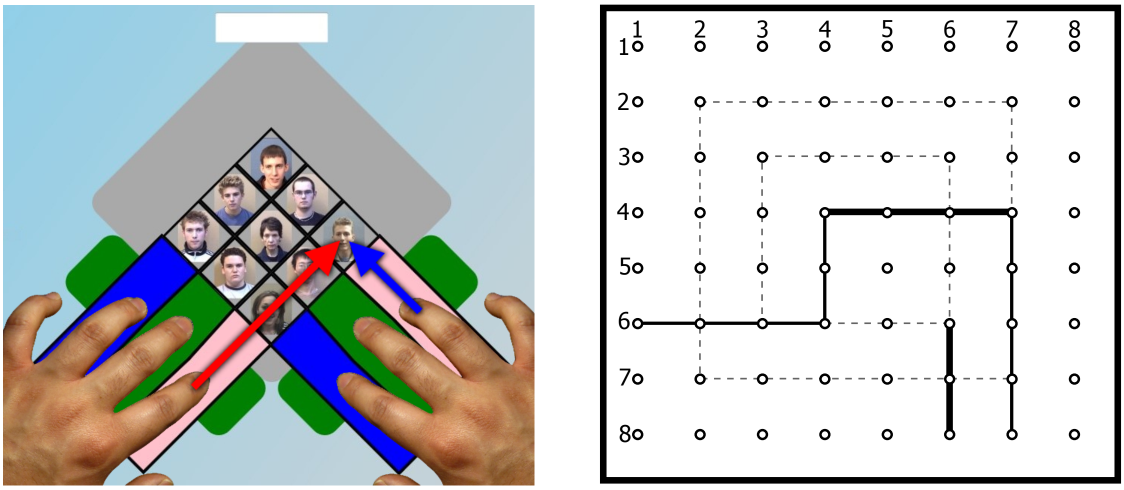


Figure 2.9: Left: PressureFaces by Kim et al. [74] which uses pressure information on a multi-touch surface for secret authentication. Right: A similar approach by Malek et al. [89] that extends drawing based graphical passwords with pressure information to secure the input. Bold lines indicate pressure.

A very illustrative example is using eye tracking technology to securely authenticate to a system as for example proposed by Hoanca et al. [61] as a security improvement for Passfaces^{TM}. The basic idea behind these systems is that the channel to the terminal, the users' gaze, is invisible and thus completely shoulder surfing resistant.

One of the first thorough approaches in this area has been implemented by Kumar et al. in 2007 [78], in which they evaluated standard gaze-based interaction techniques on their appropriateness for password-entry. Technique number one was dwell time [87], in which a user has to focus on a specific area, like a button, for a specific time to trigger an action. The second technique was called "gaze and trigger" in which an action was triggered by a button press. An evaluation using a set of alphanumerical passwords, revealed performance problems of the approach but at the same time potentially high security in combination with ease-of-use. The main problem besides performance is the need for eye tracking technology at the terminal that can precisely identify the location of the users' gaze and the need for a calibration mechanism that can cost the users significant amounts of time. The same problem applies to Cued-Gaze Points by Forget et al. [54] that applied eye tracking to Cued Click Points [17], thus requiring a user to look at specific points in a picture in a given order. In the scope of this thesis, a gaze-based authentication mechanism based on gaze gestures [43] was developed and evaluated that overcomes this weakness [35]. In a second iteration, a significant performance enhancement for this system was developed, namely EyePassShapes [27], which will be introduced in chapter 3.2.4.

The next two systems use pressure as the secret information from the user to the terminal. This is based on the assumption that pressure is an attribute that is very hard to spy on by an attacker. Theoretically, even video attacks can theoretically be overcome this way, even though none of the presented systems actually employed such an attack.

Haptic-based graphical passwords have been proposed by Malek et al. [89]. Their system uses pressure based surfaces to improve the security of "Draw a Secret"-like drawmetric passwords [67]. In addition to the users' shape, the system remembers a binary pressure information for each stroke (pressure yes or no) as shown in figure 2.9, right. This way, the secret second dimension makes attacks much harder. Unfortunately, the evaluation presented in their work is purely qualitative and thus does not allow objective judgment of the system's performance besides the fact that the study participants seemed to like it. Another weakness is that only part of the authentication credential is hidden. Therefore, based on the password length, the missing information can theoretically be identified in a certain number of rounds.

Similarly, Kim et al. [74] use pressure information on multi-touch for their PressureGrid system. Figure 2.9, left, shows PressureFaces, a PressureGrid variant based on PassFaces$^{\text{TM}}$. To select the photos that build the users' password, the users have to add pressure to the respective fingers that, in combination, uniquely identify a cell in an 3x3 grid. For instance, selecting the middle photo requires the user to add pressure to both middle fingers. The intersection of these imaginary lines marks the cell. To "force" the users to behave securely, the system only works if all buttons are occupied with a finger. To additionally hide the pressure and confuse and attacker, the buttons blink. The main advantage of this system compared to the haptic-based graphical passwords is that the whole input is hidden and thus a visual attack does not reveal parts of the authentication token. Additionally, with an average input speed of twelve seconds, the system performs well, especially compared to software based authentication mechanisms.

An advantage that both systems share is the (potential) use of multi-touch hardware, which could also run haptic-based graphical passwords. This technology is very likely to widely hit the public terminal market in the near future. In some scenarios, in which multi-touch screens are available like some public information screens, these systems could be deployed without any major additional costs.

Finally, Pass-thoughts by Thorpe et al. [127] clearly deserves to be mentioned in this chapter. In 2005, they discussed a theoretical system in which the users' thoughts could be used to securely authenticate. This is clearly usage of a hidden channel to the terminal. However, some time will pass till such a system can be effectively evaluated.

The special appeal of hardware based authentication mechanisms is their great security potential. Enabling the users to secretly receive or transmit information from and to a terminal offers great possibilities. As seen, some hardware based systems additionally manage to achieve high performance in terms of input speed. Vandalism and more specifically manipulations are their main problems alongside with potential deployment costs. Correct use of the systems can be an issue as well that might open security holes. In this thesis, two hardware based authentication mechanisms were developed and evaluated based on eye tracking which seemed to have the highest potential among the different approaches. EyePIN [35] and its usability extension EyePassShapes [27] were designed to overcome the main problems of gaze-based authentication, deployment costs and the need for calibration (see chapter 3.2).

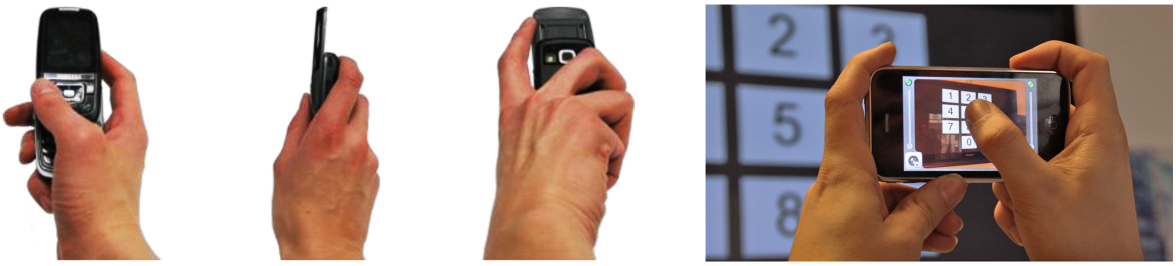


Figure 2.10: Left: A “tilt left” gesture of the gesture based authentication system by Chong et al. [18]. Right: PIN-entry using the Touch Projector system by Boring et al. [9, 10].

2.4.3 User Owned Device Based Approaches

The final category, user owned device based authentication mechanisms, is similar to the second category in that these systems require additional hardware. The big difference is that the hardware is owned by the users, already in their possession and carried around by them. A typical and often used device is the users’ mobile phone. Systems based on hardware owned directly by the user have the potential to eliminate the two main weaknesses of hardware based approaches. Firstly, it does not create additional costs for the service/terminal provider since mostly, hardware is employed that is already owned by the user. Since this hardware is not available to an attacker, it cannot be manipulated as is the case for hardware fixed to a terminal. While user hardware overcomes those weaknesses, it opens new ways for attacks. For instance, these systems often rely on wireless communication with the terminal which makes them vulnerable to man-in-the-middle attacks and establishing a secure channel to the terminal is a tricky and time consuming task that can easily annoy users.

User owned device based approaches are gaining importance with the rise of smartphones and other modern high tech devices. These often come with a huge variety of different sensors that can be exploited for authentication mechanisms. Not surprisingly, many approaches therefore rely on gesture based authentication, that is, the users perform a (visual) gesture with their devices to authenticate. However, authentication mechanisms in this category are still rare since the required hardware is just widely hitting the market.

One of the main weaknesses of gesture based passwords is described in the work of Chong et al. [18] from 2009. They explored a system based on ten different gestures (an example is shown in figure 2.10, left). Theoretical security of the system is fine and it is hard to record the authentication since an attacker has only limited knowledge about where the device will be located when the interaction takes place. However, the system is very insecure against standard shoulder surfing attacks performed by an attacker in personal. Therefore, the authors suggest their system to be used in private rather than public settings, thus, defining an explicit context.

However, the value that comes with gesture based approaches, which is exploiting the users’ muscle memory, still drives research to more secure solutions. As a result, in 2010, Kirschnick et al. [75] presented an authentication mechanism based on gestures that, in addition to the gesture itself, uses biometric information of the user to identify not only the gesture but also who is doing it. This way, a second, invisible information is used to make the system resistant to shoulder surfing attacks. In contrast to common biometric authentication mechanisms, being a secret task that the user is not actively performing, the system is not necessarily seen as a biometric system by the users.

Another category of approaches in this field relies on the unpredictability of the location of the mobile device and its small input and output hardware to render attacks, including shoulder surfing, ineffective. Most skimming attacks, i.e. manipulating ATMs, are based on the assumption that the attacker knows with 100% certainty where the input will take place. For instance, a camera pointing at the keyboard requires the input to happen at exactly that specific location to make the attack effective. This means that simply dislocating the input from the terminal improves security significantly. On the other hand, the plain input on the device might be an easy target to a shoulder surfer, which can be solved by using master PINs and other ways of clever design as for instance proposed by mobile phone authentication systems by Bianchi et al. [6].

The simplest imaginable form is doing the "plain" input on the mobile device. A system like this was for instance envisioned by Boring et al. [9] as shown in figure 2.10, right. In their approach, a keypad is displayed on a public screen. The input is done by filming the screen with the mobile phone's camera and virtually "pressing" the keys on the screen of the mobile device. Technically, the system is based on the Touch Projector prototype [10] that was built for mobile interaction with public screens. There is no evaluation available but it can be expected that it does not take significantly longer than entering a PIN on any touch screen. However, the visual search and focus tasks, by trying to film the respective part of the screen, might add inconvenience and increase input times. The risk of a real shoulder surfer also remains.

User owned device based approaches have manifold theoretical advantages when it comes to security and deployment. Costs are low and dislocating the input from the terminal makes them highly resistant to skimming attacks and other manipulations of the terminal. While real shoulder surfers and theft of the device are issues, they seem to be solvable by clever design. However, connection is an issue yet to be solved that most of the work does not cover at all. Whenever a device is about to be used with a terminal it has to be connected to it. This connection should be secure (e.g. resistant to man-in-the-middle attacks), easy and fast. Honestly measuring and reporting interaction speed therefore has to include those times as will be discussed in chapter 3.3. The connection issue is additionally highly important when it comes to defining appropriate scenarios. For instance, connection becomes a minor issue when the device has to be connected anyways if it is an essential part of the interaction. This thesis explores this field deeper than it has been done before and presents two user owned device based approaches, MobilePIN [29] and VibraPass [34] which highlight the importance of connection time and partially solve the shoulder surfing and secure transmission problems as shown in chapter 3.3.

2.5 Behavioral Factors in Authentication

Up to this point, the here presented work had a tight focus on technical aspects of authentication mechanisms. Security, usability and memorability issues are to be solved using technical means that "seem appropriate". In this thesis, it will be argued that behavioral factors or characteristic human behavior are heavily influencing security, usability or memorability. That is, they have to be considered not only during evaluation but also during the design phase of an authentication mechanism. Doing so, different standard problems of secure authentication mechanisms can be avoided early in the design process. An example that emphasizes this argument is how security can be heavily influenced by forgetting about behavioral factors. Social compatibility (or lack of it) can, for instance, open security holes when it makes users behave insecurely "on purpose" [33] as will be more thoroughly presented in chapter 5.

Behavioral factors are rather seldom discussed when it comes to the usability and security of authentication mechanisms. This partially leads to the believe that problems of authentication are either technical or user made. Though sounding obvious, Adams et al. [2] were the first to strongly advocate for the users and taking their side. They point out that considering the user the enemy, as often referred to, and treating them as such, cannot improve the security of authentication systems. They analyzed typical security holes that are opened due to users behaving insecurely. For instance, they showed that strong password policies (e.g. requiring passwords to consist of combinations of characters, digits and special characters) or user punishment (consequences arising from non-compliance to security rules) often open more holes than they close. Thus, the gap between the system designer and the users is what has to be closed to get a grip on the current situation. Instead of working against the users, they should be included into the design process of password policies and the like. They further propose several properties that should be considered to create secure and usable password systems like how users perceive a system and how to get them to understand the systems more. Coming back to the example of password policies, Florêncio et al. [52] could show that in many cases, strong password policies do not come with strong security requirements. Economical factors seem to have a much deeper impact on it. Considering the discussion by Adams et al., such decisions are hard to communicate to a user and are clearly made without taking them into account. Therefore, an important question that system administrators have to ask themselves is: "How can I expect my users to understand and accept rigorous and annoying password policies if even I do not understand where they come from and why we have them?"

The previous work focused on mistakes that are often made during the deployment and maintenance of password systems with a focus on behavioral factors. The influence of social and external factors on ATM use, that is usage of public terminals, has been approached by Little and al. in a series of work [83, 84]. The authors performed interviews with ATM users not only to find out about these factors but also about how they influence each other and which are more important than others. They include perceived levels of privacy, intimacy and security, time pressure and anxiety. Time pressure, for instance, is a very interesting factor based on the fact that users do not want to keep other people waiting. During their interviews, this kind of pressure was identified as a reason for usability problems like incorrectly entered PINs. Security is an example of a factor that is further influenced by external factors such as “time of day”.

Besides such focused work, behavioral factors are only marginally approached as a side-effect of technical evaluations. Especially long-term studies have the potential to identify such facts. For instance, as mentioned before, Davis et al. [24] found out that the security of cognometric systems (based on image recognition) can be heavily influenced by behavioral factors like personal preferences. They showed that users are more likely to choose “beautiful faces” as their authentication tokens and thus significantly reduce the theoretical password space of such systems. Based on a long-term study on biometric verification, Coventry et al. [23] showed how attitudes towards an authentication system can significantly change through usage and habituation. They also observed how negative feelings like anxiety can decrease when users are better informed about details of such systems.

Even though focused work on behavioral factors in authentication processes is still rare, the few existing examples highlight an important fact: considering these factors in the design and evaluation process of an authentication mechanism (and also during maintenance), has the potential to both reduce user caused security holes and increase acceptance and overall security of the system. Therefore, dedicated parts of this thesis were conducted with the only goal to gather criteria of behavioral factors and use them to improve the effectiveness of authentication mechanisms as well as their design and evaluation. The most important part of this thesis with respect to such factors is thus chapter 5, which presents a field study on ATM use. Additionally, behavioral factors were derived from the work on long-term evaluations as reported in chapter 4 as well as during the work on novel authentication mechanisms which are discussed in more detail in chapter 3.

2.6 Lessons Learned

Analyzing related work and specifically how and based on what criteria authentication mechanisms are evaluated, as outlined in this chapter, several problems become apparent. One of the most prominent ones is that it is hard to compare authentication mechanisms to each other since the information provided by the authors is sometimes too limited and the approaches on evaluation vary heavily. For instance, time is measured from many different perspectives, sometimes forgetting important parts like phases before the first button press is performed that should be part of the measurement. Some papers report on learning phases, some do not. In this case, this can mean that some reported times are actually learning phase times while others are based on trained users, which additionally makes comparison of input times very difficult. Another point, that is approached from many different angles, is security. While most work provides a theoretical security analysis, real proof of security based on skimming or shoulder surfing attacks is often neglected. Within this thesis, this problem will be approach by providing extended and new criteria for the creation of authentication mechanisms for public spaces.

Another problem is that, too often, no discussions are provided on which scenarios the respective authentication mechanisms are meant and appropriate for and why. This is problematic since scenarios can be important factors to justify which approach (software, hardware or user owned device based) has been chosen for the design of the system. For instance, one of the biggest disadvantages of user owned device based authentication approaches, connection effort and time, does not play an important role anymore, if in the scenario, a connected device is required anyways.

One of the major findings of the analysis and categorization of related work is about where the respective approaches require improvement and where their main weaknesses come from. This way, the systems presented in this thesis can build on this analysis and try to overcome the identified problems or at least partially solve them. The here presented categorization seems the most appropriate with respect to possible scenarios, advantages and disadvantages.

Finally, the lack of knowledge on behavioral factors and their influence on authentication are alarming. Even though all related work that took such factors into consideration could show how they significantly influence positive or bad performance of an authentication system, they are too often not taken seriously. Believing in the potential of behavioral factors, they will play an important role for the creation of criteria within this thesis.

Chapter 3

Learning from the Design of Authentication Mechanisms

I haven't failed, I've found 10,000 ways that don't work

– Thomas Alva Edison –

We developed five authentication mechanisms, presented in this chapter, that were designed to cover the different categories discussed in the related work chapter 2. Design, implementation and evaluation of these systems contributed to this thesis in the following ways:

a) It helped to identify weaknesses in the state-of-the-art evaluation process of authentication mechanisms. Based on this, standard usability and security criteria could be extended and finer grained and new criteria were found. Some of the findings, like how (or better not) to employ user "skills" to secure authentication, could not have been made without practically implementing and testing real systems.

b) The here presented systems solve problems of authentication mechanisms in their respective categories and thus present candidates that – considering specific improvements – could solve the problem of different scenarios of public authentication.

c) The evaluation of the authentication mechanisms uncovered knowledge gaps, especially on behavioral factors, that, as a consequence of this work, were approached in additional long-term and field studies.

3.1 Threat Model

An often forgotten, but extremely important step in the design and evaluation of an authentication mechanism is to know what a system should provide security against. Thus, defining an appropriate threat model has to be an essential part of the design process of any secure system. It is important to define the resources available to an attackers and describe their expertise. The here presented threat model builds the foundation of the evaluation of the authentication mechanisms proposed in this chapter. Due to the fact that some come with additional threats, like man-in-the-middle attacks that only apply for user owned device based approaches, this model will be slightly extended in the respective sub-chapters.

We suppose an attacker that has access to all standard attacks as known from ATM frauds. These attacks include skimming devices like card readers and fake keypads as well as simpler attacks like shoulder surfing, that is, an attacker standing close to the victim and trying to spy on the input. Furthermore, we assume that the attacker has thorough knowledge of the respective authentication mechanism and knows its weaknesses as well as the most appropriate attacks. This will be referred to as “worst case” scenarios in the scope of this thesis. We furthermore suppose that the attacker can employ different kinds of attacks that try to exploit weaknesses of the theoretical security of a system as well. Examples are educated guessing attacks as well as dictionary attacks.

Additionally, the here defined threat model is based on the special conditions that come with designing secure systems for public spaces. This means that the attacker has full access to the public terminal at which the authentication will take place. The attacker can add arbitrary hardware to the terminal to commit an attack. For instance, the previously mentioned card reader can be used to copy the user’s bank card or hidden cameras can be utilized to record all of the user’s input. For standard PIN-entry, such an attack usually results in the attacker gaining possession of the credentials necessary to withdraw money. For other systems, different hardware based attacks have to be assumed, for instance, audio recordings. Attacks like shoulder surfing are also more prominent and dangerous due to the public setting.

3.2 Hardware Based Approaches

In contrast to the order in the related work chapter, which started with started with software based systems and ended with user owned device based approaches as the newest “trend”, the systems in this chapter will be sorted starting with hardware based approaches and ending with software based approaches. This order represents the closest possible approximation of the chronological order the system were developed in.

Based on related work, one special direction of hardware based approaches seemed to be very promising: eye tracking technology. Being a system that secretly receives information from the user on an invisible channel, its security properties are very high. Thus, the focus of the eye tracking based authentication mechanisms in this chapter was to overcome the two main weaknesses of eye tracking based authentication, the need for calibration and expensive hardware (and thus high deployment costs).

3.2.1 A Concise Introduction to Gaze Gestures

EyePIN [35] and EyePassShapes [27, 37] are two authentication mechanisms using gaze gestures that we have designed and evaluated within this thesis. They are based on recent advances in the field of eye tracking technology and more specifically, gaze based interaction. Therefore, this chapter provides a quick introduction to eye tracking with a focus on gaze gestures.

Gaze based interaction techniques is a rather old field of HCI. Techniques like dwell time have already been evaluated in the late 80s and early 90s [66] and were originally meant to support disabled people in interacting with computing devices. Interaction with dwell time means that the user stares at a point for a specific time to invoke an action. In this context, especially eye typing [88] has gained a lot of attention. Usually, eye typing uses dwell time and on-screen keyboards. That is, the user types in characters by concentrating on them on the screen for a specific amount of (dwell) time. In recent research and also in the advertising business, however, eye tracking technology is increasingly used as a tool to test the usability of systems or attractiveness of ads since it allows the researcher to get a deeper understanding of attention spans of their users and the like [72]. Standard interaction and usage scenarios for eye tracking thus rely on knowing the (almost) exact position on the screen the user is looking at (e.g. [73]). Therefore, good (and expensive) eye tracking hardware is required in addition to calibration of some sort.

One of the newest concepts of eye tracking are gaze gestures. The idea is to do input by performing specific shapes (gestures) with the eyes. This means that neither do gaze gestures require and block screen real estate (since there is no interface) nor do they require calibration. The lack of calibration has a simple explanation: to work properly, the gaze gesture algorithm does not have to know the exact position of where a user is gazing at but only relative movement based on relative coordinates. Therefore, gaze gestures are a cheap and highly error-resistant gaze based input technology.

For a long time this approach has been neglected due to the assumption that controlled eye movement is a rather hard to perform and cumbersome task. Drewes et al. [43] were the first to show that using gaze gestures for specific input tasks actually can make sense and that users can intentionally create these shapes using their eyes. They conducted different experiments and found that for simple tasks like switching an audio player on and off or forwarding songs, gaze gestures can be an appropriate tool. During the work on hardware based authentication based on eye tracking, these results were a big encouragement to utilize gaze gestures for authentication purposes.

Also in other areas, gaze gestures finally got more attention. Thus, researchers performed thorough analyses of interaction systems based on gaze gestures. For instance Wobbrock et al. [137] evaluated text input based on gaze gestures while Bulling et al. [14] even created an eye tracking

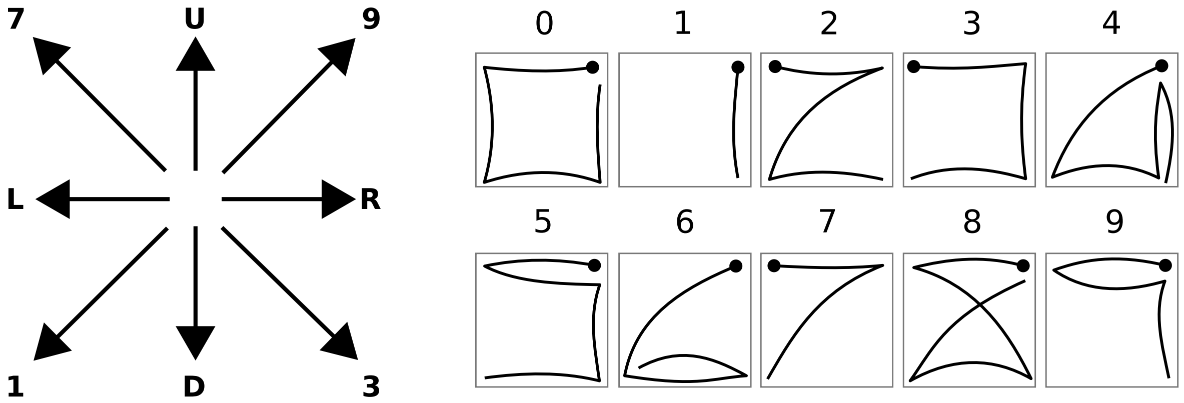


Figure 3.1: Left: Gaze gestures as in the concept by Drewes et al. [43] consist of arbitrary combinations of eight different strokes that perfectly fit the physiology of the human eye. The numbers and letters are the internal representation of the strokes. Right: The ten different gestures used for EyePIN. The digits are an eye-optimized version of the EdgreWrite alphabet by Wobbrock et al. [136].

device which, due to its nature of recording relative movements only rather than exact points, perfectly fits the gaze gestures concept. These results added further value to the assumption that gaze gestures for authentication purposes actually do make sense.

3.2.2 EyePIN

Being a more or less invisible interaction, gaze gestures are an interesting candidate for an authentication mechanism. Following this idea, we designed and evaluated the EyePIN system [35][1].

Concept

The concept of EyePIN strictly follows the gaze gestures approach as developed by Drewes et al. [43]. Thus, a gesture consists of an arbitrary combination of the nine different strokes shown in figure 3.1, left. Designed for PIN-entry, the EyePIN system only requires ten different gestures to work, the digits zero to nine (figure 3.1, right). The gestures for the digits are based on the EdgeWrite alphabet presented by Wobbrock et al. [136]. Small modifications have been made to better fit the properties of gaze entry.

The one major difference in comparison to the original concept by Drewes et al. is the use of a dedicated button to trigger gesture recognition. In the original concept, the user's gaze is continuously analyzed in the search for gestures. EyePIN on the other hand only analyzes the gaze as long as a specific button is pressed. This has several advantages. In the work of Drewes

[1] This chapter is partially based on a conference paper that we published in 2007 [35]. Some of the data was analyzed again for this thesis to get further insights and to better relate them to the other authentication mechanisms that we developed.

et al., gestures have to be significantly different from any naturally occurring gesture so no false positives occur. Using the gesture button, this limitation does not exist and thus gives more freedom in the design of the gestures. For instance, the simple gestures used for 0, 1 and 3 (figure 3.1, right) are not possible in the original concept since these gestures regularly appear in the users' normal gaze, e.g. while reading [43]. For instance, a square (the digit 0) has been identified as naturally occurring during reading tasks, even if the texts are short. The gesture button used in the EyePIN prototype was the space bar. Thus, to input a digit of the PIN, the user has to hold down the space bar and perform the digit's gesture with the eyes. This has to be repeated four times to enter a four-digit PIN.

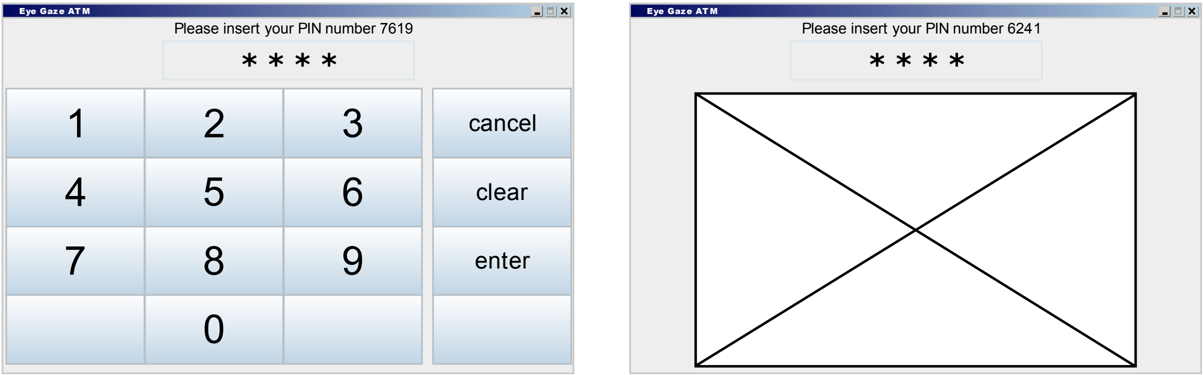
The main advantages of EyePIN qualify it for a scenario involving a public terminal like an ATM. First of all, EyePIN does not require a calibration process since a gaze gesture uses relative movements only. Also accuracy is not an issue as a gesture is several degrees of visual angle. This means that a system that supports EyePIN requires only low-tech eye trackers. This could be a normal webcam combined with an infrared LED light, which produces the red eye effect, known from photographs that simplifies the pupil recognition due to the higher contrast against the iris. Considering that many ATMs are already equipped with cameras (e.g. for security reasons), deploying EyePIN only requires simple and cheap hardware. The second main advantage of EyePin is that there is no need for interaction objects on the screen.

Prototype

To test the feasibility of the EyePIN concept, it was not only important to evaluate the system itself but more importantly to see how it performs compared to standard eye tracking input techniques. Therefore, we implemented three different interaction techniques for the user study: (1) EyePIN, which uses gaze gestures, (2) a system based on "dwell time", and (3) a technique that in the scope of this work will be called "look & shoot". The latter two represent the most dominant eye tracking interaction techniques which makes them the best candidates for control groups. Therefore, the prototype implementation follows the experiments conducted by Kumar et al. [78] with the extension of using EyePIN, a gaze gestures based approach. "Dwell time" and "look & shoot" will be further defined in the following.

"Dwell time" has the advantage of being a very straight forward and easy to understand interaction method. Staring at a specific point on a screen for a specific amount of time to trigger an action is a very simple and intuitive task. The success of this method significantly depends on the chosen dwell time. Short dwell times allow for fast and fluent work but are more likely to produce errors by unwanted fixations, also called the Midas-Touch effect [66]. Long dwell times slow down the interaction. Within this work, a dwell time of 800ms was used in the experiments. This value showed promising results in pre-tests.

"Look & shoot" is another technique that we borrowed from classical eye tracking. In this approach, the user fixates an object on the screen and simultaneously hits a button to trigger an action. In the focus of these experiments, the space bar was used as the trigger button since it is rather big and easy to reach. The main advantage compared to dwell time is that, besides being intuitive and easy to use as well, it is faster based on the fact that no dwell time has to elapse



Eye Gaze ATM

Please insert your PIN number 7619

* * * *

Eye Gaze ATM

Please insert your PIN number 6241

* * * *

Figure 3.2: The user interfaces of the EyePIN study. Left: The same user interface was used for the "dwell time" and "look & shoot'" interaction techniques. Right: A grid is displayed to give some visual support for the users of the EyePIN authentication mechanism.

before an action is triggered. Using exact locations on the screen for interaction, both dwell time and look & shoot require a calibrated eye tracking device. In contrast to those, EyePIN does neither require calibration nor absolute positions.

The prototype for the three different interaction techniques was written in Java and was running on a Windows PC using a commercial eye tracker (see figure 3.3). The user interface for the dwell time and look & shoot techniques displayed an ATM number pad as known from standard keypad layouts. This interface is shown in figure 3.2, left. The number pad had an overall size of 730x450 pixels. Each button had a size of 180x190 pixels, which is about 5° to 3° visual angle and therefore clearly above the typical eye tracker accuracy of around 0.5°. To input a digit, the users had to gaze at the respective number and trigger the action by using the space bar (look & shoot) or waiting (dwell time). The interface for EyePIN consisted of a black cross on a white surface (figure 3.2, right). The cross, as well as the corners of the background were meant to provide anchor points for the users' gaze to make performing the gestures more easy. However, it was not required to use the cross since any relative movement was accepted by the prototype.

The only visual feedback given were asterisks that highlighted whether a digit was input or not. Highlighting techniques – like letting a software button glow when it is in the user's gaze – were not used since they would decrease the security of the system.

Theoretical Security Analysis

As stated in the threat model, skimming and shoulder surfing are the most common methods used by criminals to get in possession of a person's PIN. Eye-gaze interaction as an input method is completely resilient against shoulder surfing attacks, since observing the movement of the user's pupil is impossible and will definitely arouse suspicion.

A camera recording of the eye movements on the other hand, can significantly help the attacker. A difficulty for the attacker is that this camera must necessarily be placed in the field of view of the user, which makes it harder to conceal it. Additionally, determining the actual PIN from the raw data is not a trivial task. Especially filtering the eye-input sequences from "normal" eye movement caused by reading can be difficult. A successful attack against look & shoot or EyePIN additionally requires synchronous recording the trigger button. Thus, using a hardware button can be credited an extra security benefit. The raw eye movement data itself can only be used for extracting the PINs with an enormous effort by the attacker. Therefore, it is reasonable to assume that using eye gaze input will prevent the vast majority of shoulder surfing and skimming attacks due to increased complexity. The attack that promises the best results would be to add an eye tracking device to the public terminal.

In this spirit, being the advantage of EyePIN, the lack of calibration can turn out to be a security risk since not only the hardware required for the ATM to work but also the hardware that would make an attack work is much cheaper in production and more easily deployable.

Since all three systems are based on PIN as an authentication token, they inherit its security properties, the good as well as bad parts. That is, the password space, possibility of dictionary and educated guessing attacks and other weaknesses/strengths.

Evaluation

Figure 3.3 shows the user study setting with which EyePIN was evaluated. The setting consisted of the eye tracker which was equipped with the prototype software. To monitor the experiment, a second screen was used to display debugging messages and the like for the study observer. Since expecting from the users to remember all EyePIN gestures seemed impossible within the scope of the study, the complete gesture-to-digit list was provided to the users and always visible next to the eye tracker. Finally, due to the extreme sensitivity of the employed eye tracker, a chin rest was set up at the desk to support the users interacting with the system. This was necessary for the dwell time and look & shoot interaction techniques since they are very sensitive to calibration shifts. This was mostly based on previous experiences with the technology.

User Study Design EyePIN was evaluated against the other techniques using a *repeated measures within participants factorial design*. The independent variable was *technique* (dwell time, look & shoot, EyePIN). The task was to authenticate to the terminal using every technique with three randomly generated PINs (technique x 3 = 9 authentication sessions per participant). The order of the techniques was randomized to minimize learning effects.

Procedure and Participants At the beginning of the experiment, each user had to draw a number from a bowl, which was used as the anonymous ID for the respective user during the tasks and for the final questionnaire. This way, the analysis of the data could be performed anonymously. Additionally, the ID was used to assign a random task order to the participants. Following this step, the prototypes were explained in detail to the participants. Since eye tracking was a novel interaction technique for the majority of the participants, an extensive training sequence was provided for each technique in order to get the participants accustomed to eye tracking and the PIN-entry techniques as well. In this learning phase, they had to enter the digits from zero to

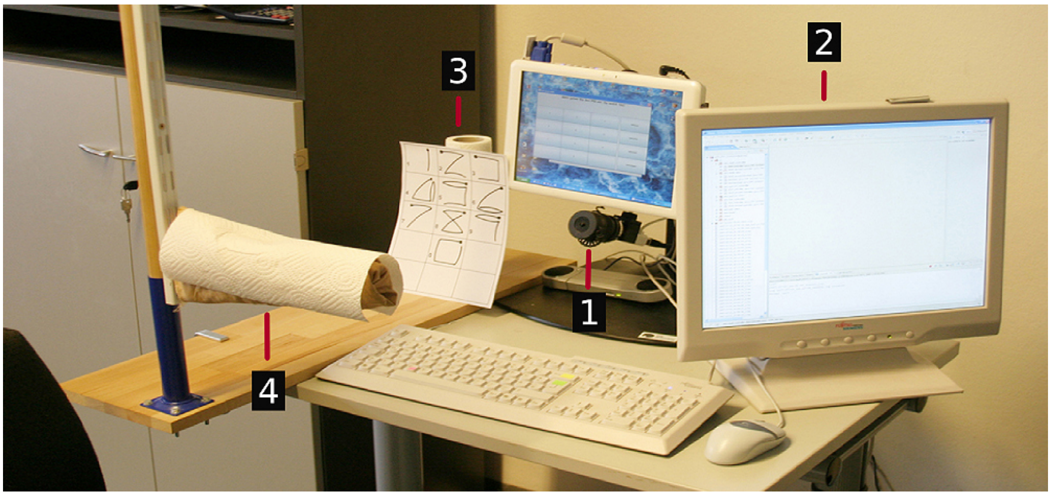


Figure 3.3: Setting of the EyePIN user study: 1. Eye tracker. 2. Study observer monitor. 3. A list with the gestures for the digits was provided to the users. 4. Chin rest for the dwell time and look & shoot methods.

nine sequentially with each authentication mechanism. For the main tasks, the participants had to enter three randomly generated four-digit PINs with each interaction technique. For each PIN, they had three tries to successfully enter the PIN. Directly after the practical part, demographic data, information about the users' habits using ATMs and data regarding user experience and usefulness of the systems were collected using a post questionnaire.

The study was conducted with 21 volunteers. All participants were between 22 and 37 years of age, seven female, the rest male. Only five participants had prior experience with eye tracking devices. Therefore, the results of the evaluation are based on 63 authentication attempts (21 x 3 PINs).

Hypotheses For the EyePIN user study, the following hypotheses were stated based on the assumption that EyePIN takes a specific minimum time due to biological constraints of eye movement:

(H1) EyePIN is the slowest method.

(H2) EyePIN is less error-prone than the other methods.

Results

Authentication Speed Input time was measured from the first key press (entering the first digit of the PIN) to the final confirmation of the PIN (pressing the enter button on the keyboard). Being one of the first studies on authentication that we ever performed, we did not log detailed information on the different stages of interaction. For instance, there is no information in the log files to identify the time it would take without pressing enter. As will be explained later in this

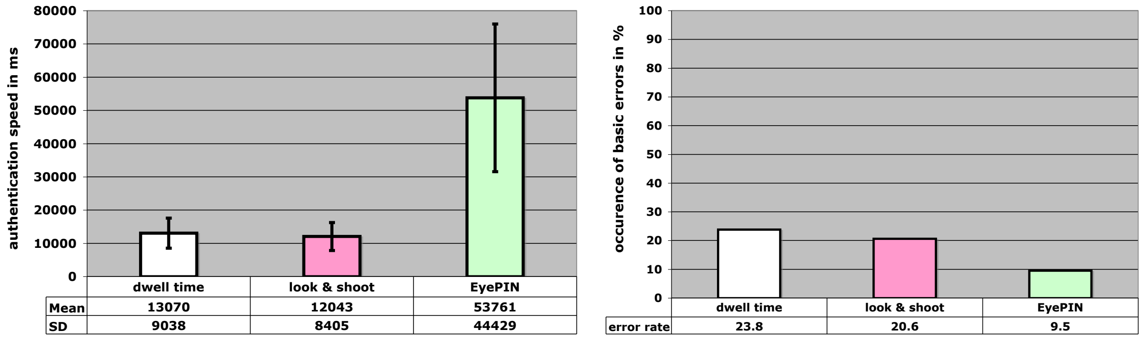


Figure 3.4: Results of the EyePIN user study. Left: Average authentication speed of the EyePIN user study. Dwell time and look & shoot performed equally while EyePIN was the slowest. Right: The tendency for the error rate, defined as the appearance of basic errors, has the opposite tendency with EyePin performing best.

thesis in chapter 4, presenting time this way only does have its downsides. For instance, pressing enter takes the same amount of time no matter which method is used and can thus be considered an unfair advantage for slower systems. EyePIN, being expected to be slower, does thus have such an unfair advantage. This means that the mean values presented in figure 3.4, left, include the time required for confirmation. As opposed to other studies in this thesis, the time includes failed attempts. As a result of this study, we understood that we can learn more from a study, if we analyze these parts separately.

The results show that EyePIN was the slowest method (M=53.76s, SD=44.43s). Dwell time (M=13.07s, SD=9.04s) and look & shoot (M=12.04s, SD=8.41s) performed almost equal. We have to note here that all three systems perform worse than what we are used from standard PIN-entry. A one-way repeated measures analysis of variance showed a highly significant influence of the authentication method on authentication speed ($F_{1.068,66.23}$=53.084, p<.001). A post hoc test revealed that the differences between EyePIN and the other two systems are highly significant (both p<.001). The difference between look & shoot and dwell time is not significant (both p>.05). These results support hypothesis (H1).

The gaze gesture method showed to be less intuitive than the classic methods. Many subjects initially had problems to produce recognizable gestures. Furthermore, the gesture alphabet was unknown to all participants. This explains the big difference in time for completing the PIN entering task. Much time was used for looking at the sheet that showed the gestures for the single digits. Theoretically, a stroke within a gaze gesture needs about 500ms (200ms for the saccade and 300ms for the fixation) and a digit with four strokes takes about two seconds to perform. A four-digit PIN with one second break between the inputs of the digits would therefore take a little bit more than ten seconds. Indeed we had a participant in the study, who entered the PIN correctly within 14 seconds. We expect times closer to this value for all users once they are trained for gaze gesture input. Nevertheless, even this speed would be significantly worse than standard PIN-entry.

Error Rate As introduced in chapter 2.2, authentication mechanisms typically differentiate between two kinds of errors. Within the evaluation of EyePIN, we used this approach, counting basic and critical errors. A basic error was defined as one or two failed attempts to enter a PIN, a critical error means that within three attempts the user could not authenticate to the terminal successfully. This distinction is based on the fact that public terminals confiscate the user's bank card in such a case.

Surprisingly, during the whole study, not a single critical error occurred. That is, the participants performed a successful authentication with the terminal in at least two tries. However, a couple of basic errors occurred. Using dwell time, 15 of the 63 entered PINs were entered wrongly at least once (23.8%). Using look & shoot, 13 contained errors (20.6%). EyePIN, despite taking much longer than the "classic" methods, was at the same time much more robust against errors. Only six out of the entered PINs using EyePIN were erroneous (9.5%). This can be credited a great advantage of the gaze gesture method, especially in regard of the fact that most PIN-authentication systems block access after three incorrect attempts due to security reasons. These results are depicted in figure 3.4, right, and support hypothesis (H2).

Explaining this effect is quite simple. If an error occurs during gesture input, the gesture will not be recognized in most cases since the digit gestures are highly distinct from each other. That is, a gesture is either correctly recognized or no input takes place at all; a simple all or nothing principle. If, on the other hand, an error occurs using the look & shoot or dwell time method, this inevitably leads to inputting a wrong digit, which cannot be registered by the user due to the lack of feedback. Keeping this in mind, it is not surprising that the six erroneous inputs with EyePIN were due to the few digits that are similar like "5" and "9" whose distance is only one stroke (see figure 3.1, right).

Discussion

Studying related literature, eye tracking stands out in the masses of hardware based authentication mechanisms having only two major disadvantages: calibration and deployment costs. Gaze gestures were specifically designed to overcome exactly these problems and thus are the obvious choice to be exploited for authentication. The work on EyePIN was a straight forward approach – using PIN as the authentication token – and a first look into if and how gaze gestures can be appropriate for authentication. Therefore, this work can be seen more as a first exploration than an intense evaluation of the concept. We performed thorough exploration in comparison to PIN within the scope of the work on EyePassShapes which will be outlined in chapter 3.2.4.

These first results, however, are very encouraging. In addition to the absence of a calibration process, the second main advantage of EyePIN is its robustness to input errors. Due to the abandonment of feedback (for security reasons), each wrong gaze leads to an incorrect PIN-entry if the dwell time or look & shoot methods are used and thus to higher error rates. Using EyePIN, a wrong gaze leads in most of the cases to an unrecognizable gesture, thus the users implicitly get feedback that an input error occurred and can repeat the input even without feedback in the classical sense.

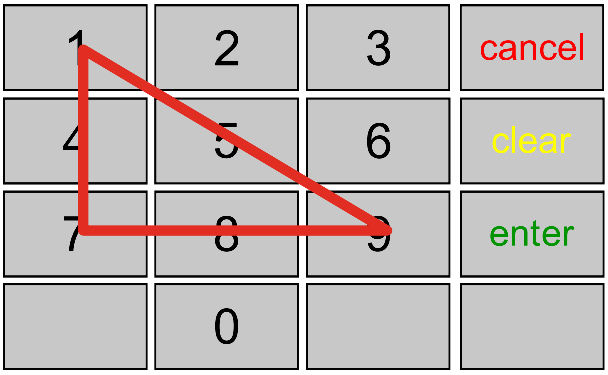


Figure 3.5: A commonly applied memory aid to remember PINs is to memorize a shape that, laid over a standard keypad, reveals the PIN. This way both, the visual and the muscle memory are utilized. This example shows a typical visual aid for the PIN "1971".

In general, the doubts that the lack of feedback would make the prototype very difficult to use were resolved in the end. None of the users even mentioned that using the systems without knowing what was entered was confusing them. The classic methods additionally benefited from large target sizes that made interaction easy.

There is a lot of open space to improve the usability of EyePIN. The gesture recognition algorithm of the prototype was very straight forward. The main reason why a gesture performed by a user was not recognized by the system was due to lack of exactness in the hand-eye coordination. As a button has to be pressed and held while performing the gesture, an additional stroke can be detected directly before or after the real gesture. These unintended upstrokes or tails could be filtered out by the algorithm and improve the recognition rate significantly. Another approach would be an adapted design of the gestures. If the "distance" of any gesture to any other is at least three strokes, the algorithm could auto-correct gestures with a single false stroke.

Unfortunately, the authentication speed of EyePIN was rather disappointing. Even though it can be considered fair to assume that users can perform much better with the system, there is a natural limit of around eleven seconds that is very unlikely to be beat. This number is based on the fact that one stroke takes around 500ms (200ms for the saccade and 300ms for the fixation), which sums up to a little more than ten seconds in average for a four-digit PIN. In this notion, an extension of EyePIN has been developed, EyePassShapes, that replaces the authentication token and thus can perform much faster. Additionally, EyePIN still lacked a practical in-depth security analysis, which will be reported in next chapters.

The First Step Towards EyePassShapes

As a side effect of the preliminary evaluation of EyePIN, we found that many participants stated to remember their PINs as a shape on the number pad rather than the actual number sequence as shown in figure 3.5. Thus, we conducted an online survey with 86 participants. The results show that more than 40% of the participants stated the same: that they would remember their PINs by a shape or at least use a shape to support their memory.

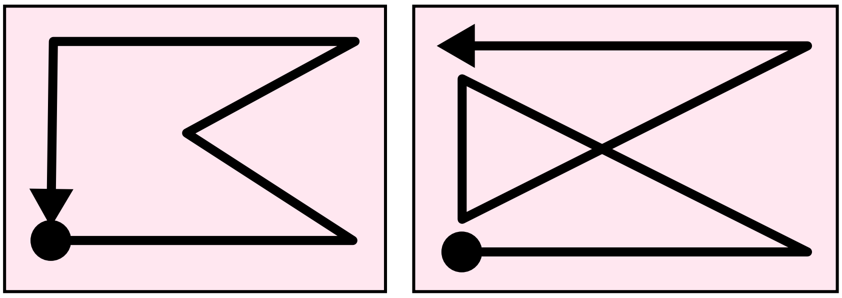


Figure 3.6: Examples of two different PassShapes as used in the user study of EyePassShapes. Left: “R79LD”. Right: “R7D9L”.

Mainly based on these results, we developed an enhancement of the gaze gesture input method for secure authentication. The idea is to improve the algorithm to enable the input of arbitrary shapes or combination of shapes and thus replace the complex digit gestures. Based on lessons learned from drawmetric systems, as introduced in chapter 2.3, the assumption is that shapes are by far easier to remember even if very complex shapes are chosen (that is, much harder to copy by an attacker). Thus, the authentication system introduced in the next chapter tries to overcome the weaknesses of EyePIN by choosing PassShapes as authentication tokens while keeping its advantages. Therefore, it can be considered a combination of EyePIN and PassShapes [36, 132].

3.2.3 PassShapes

Before we can talk about EyePassShapes, we have to introduce PassShapes. We developed the concept in 2007 [36, 132] as an alternative to *draw a secret* [67] with the main advantage (and limitation) that it relies on a simplified set of strokes that does not allow for as complex shapes as *draw a secret*, but at the same time was supposed to improve memorability.

To authenticate to a system, the users paint shapes – consisting of strokes – in a predefined order. The strokes are the same as used in the original gaze gesture concept as depicted in figure 3.1, right. A PassShape consists of an arbitrary number and combination of these strokes. The number of strokes heavily influences theoretical security and password space of the system. Internally, PassShapes are represented as a string, which makes them appropriate for standard security mechanisms like hashing. That is, they do not have to be stored in clear text on a server. For instance, the internal representation for the PassShape for up, right, down, left (a square) would be *“URDL”*. More complex five-stroke gestures are depicted in figure 3.6.

Theoretically, increased memorability is achieved in two ways, as known from drawmetric systems. Firstly, the authentication tokens are based on shapes which are essentially pictures. Thus, the pictorial superiority effect [122] – simply speaking, pictures can be more easily remembered than abstract tokens like numbers – should increase memorability. Second, PassShapes are always and repeatedly drawn in the same way following a specific order. Thus, muscle memory effects [117] positively influence their memorability. In a longterm user study, advantages of PassShapes in combination with repeated writing strategies were revealed[2].

The main disadvantage of PassShapes is that it does not increase security compared to PIN or password entry. Whenever a user draws her PassShape, nearby onlookers can easily steal it. Thus, most of the attacks that work on PINs and passwords can be used for PassShapes as well. More precisely, the same effects (muscle memory, pictorial superiority) that make the system easier to remember, theoretically decrease security since these advantages apply to an attack as well.

Summarized, as opposed to EyePIN, PassShapes is easy to use, easy to remember and fast. Unfortunately, it is highly insecure. Combining both concepts, i.e. using PassShapes as an authentication token for gaze gesture authentication, was considered to have the potential to combine the good properties of both approaches and create one secure and easy to use system, EyePassShapes.

3.2.4 EyePassShapes

Based on the experiences with EyePIN [35] and PassShapes [132], we developed EyePassShapes [27, 37] as a combination of both approaches incorporating their respective positive aspects. Just like EyePIN, EyePassShapes falls into the category of hardware based authentication systems. The difference is the authentication token. Instead of relying on a standard PIN, it employs PassShapes. As a result, it is more secure than PassShapes and supposedly more usable, and much faster than EyePIN. At the same time, it keeps the advantage to be cheaply deployable[3].

Concept

As mentioned before, EyePassShapes extends and improves two authentication methods by combining them. This way, their flaws are eliminated and replaced with the advantages of the respective other system. The two systems are PassShapes [132] and an authentication system based on gaze gestures, EyePIN [35].

EyePassShapes, uses the stroke based authentication tokens of PassShapes and combines it with the secure eye tracking approach of EyePIN. Fortunately, the strokes used for PassShapes perfectly fit the biological constraints of the human eye, which moves in saccades and cannot perform any non-linear movements. This means that the original PassShapes concept did not have to be adapted in any way to be appropriate for gaze gesture input.

Just like PassShapes, EyePassShapes can be performed in one time but also as a row of consecutive shapes (release the control key and press again). That is, the single strokes of the authentication token can be added one by one, which makes interaction easier but at the same time slower.

[2] For more information about PassShapes and its evaluations, please refer to [132].

[3] This chapter is partially based on two papers that we published in 2008 and 2009 respectively [27, 37].

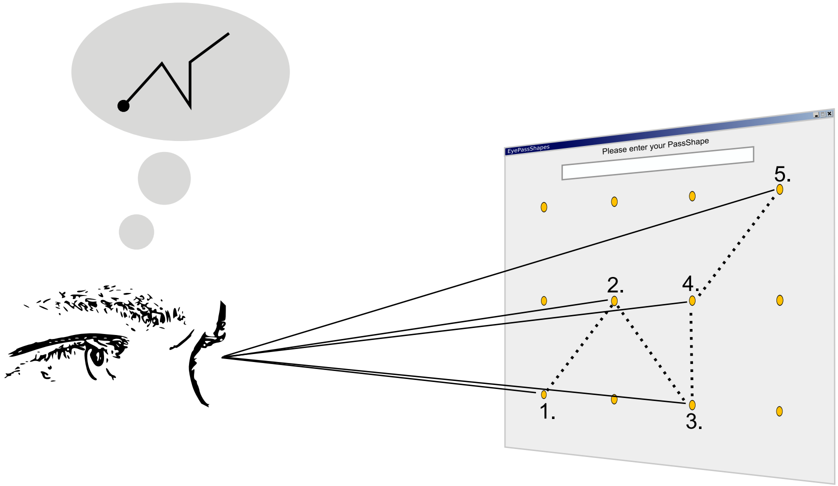


Figure 3.7: EyePassShapes uses gaze based shapes to authenticate to a public terminal. The user holds down a trigger button and performs the PassShape with the gaze. This figure shows the exemplary four-stroke gesture “93U9”.

Authentication with EyePassShapes works as shown in figure 3.7:

1. To enter the PassShape, the user holds down the trigger button.
2. Whenever the button is released, the movements that have been done during this period by the user’s gaze are analyzed. It can either contain the whole PassShape or parts (single strokes) of it. In the latter case, this step has to be repeated until the whole PassShape has been entered.
3. After entering the whole PassShape, the user ends the authentication attempt by pressing the dedicated “ok”- button (not done by gaze).
4. Finally, the shape (or the combination of all if the user pressed several times) is compared to the PassShape in the database. If they match, the authentication approach was successful.

By relying on easy-to-use authentication tokens (PassShapes), EyePassShapes is easier to use than EyePIN. At the same time, it is theoretically more memorable than standard PIN and password (and thus also EyePIN). An interesting question is whether muscle memory effects can be observed when PassShapes are entered with the eyes. At the same time, using an eye tracking input method makes it more secure than standard PassShapes due to increased shoulder surfing resistance. Figure 3.6 shows two examples of five-stroke PassShapes that could be used in the EyePassShapes system.

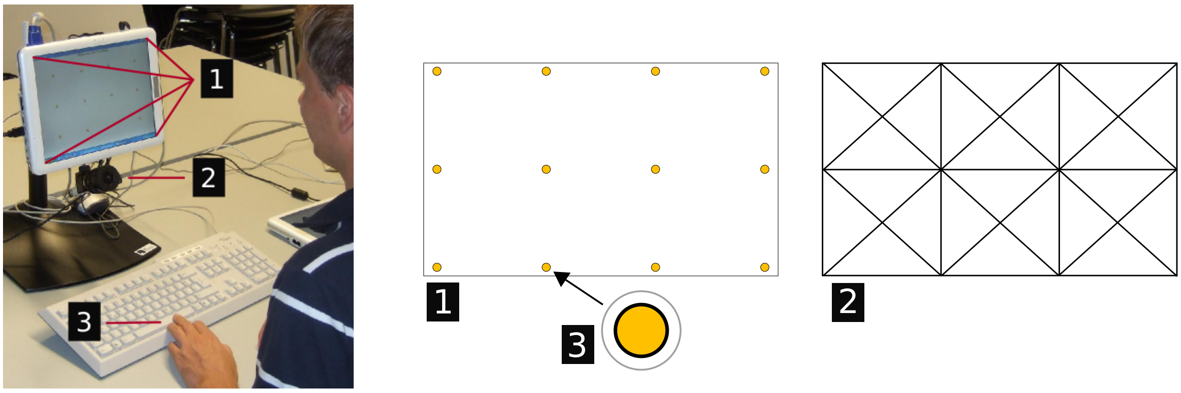


Figure 3.8: Left: EyePassShapes prototype. 1. Field of vision. 2. Eye tracker. 3. Trigger button. Right: The two different candidate background designs for EyePassShapes. 1. Dotted background. 2. Grid. 3. Magnification of a dot. After a preliminary technical evaluation, the dotted background was identified as the user interface for the user study.

Prototype

The prototype of EyePassShapes was running on the same hardware as EyePIN (see figure 3.8). As for EyePIN, the space bar was chosen as the trigger button. The EyePassShapes software was written in C++ (proxy to the eye tracker) and JavaSE (gesture recognition and user interface).

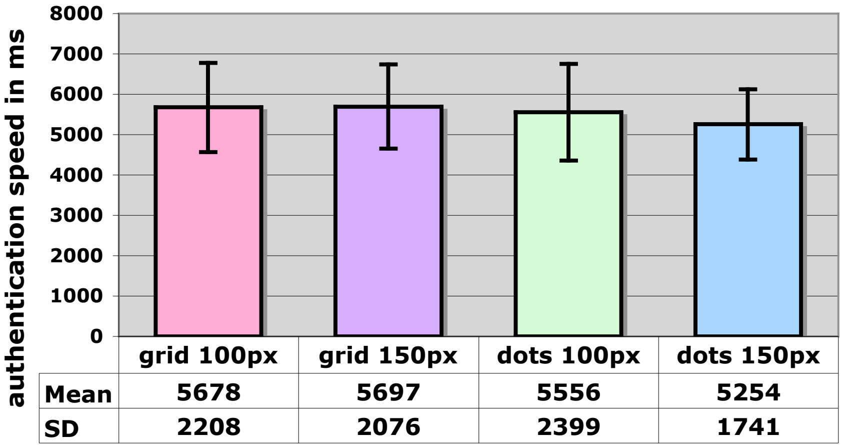
To find the right settings for the EyePassShapes user interface and software, we conducted a preliminary user study. The evaluation started with one main question: *Should visible aids be provided and if yes, of which kind?*

This question refers to the choice of a background image. Should the background provide visual aides at all? In the work of Drewes et al. [43], the authors let users perform very simple gestures and tested them on different backgrounds. They chose a blank background, a spreadsheet (a work environment) and a grid background. Surprisingly, with all designs (even with the blank screen) users performed rather well. However, a deeper look at the data revealed that for the blank screen users created themselves visual aides. For instance they used the screen corners or stains on the screen. These tricks worked fine for the simple gestures used in [43].

Since the PassShapes used for EyePassShapes are slightly more complex than the gestures tested by Drewes et al., a more advanced evaluation of the backgrounds for the final prototype seemed appropriate. Informal evaluations showed that only advanced EyePassShapes users were able to perform the shapes on a blank screen. In the end, two possible designs remained as depicted in figure 3.8, right: One consisting of simple points, and the other one depicting a grid design similar to the one used in [43]. Both designs enable the input of shapes with a horizontal span of three and a vertical span of two strokes, which allowed for the widest distribution of strokes considering the technical constraints of the used eye tracker. That is, PassShapes that fit in this area can be performed as a single stroke. If a shape requires more horizontal or vertical space, it has to be performed in several consecutive steps.

Table 3.1: Numbers of authentication attempts that failed during technical evaluation.

grid 100px	grid 150px	dots 100px	dots 150px
4/17	3/17	0/17	2/17



	grid 100px	grid 150px	dots 100px	dots 150px
Mean	5678	5697	5556	5254
SD	2208	2076	2399	1741

Figure 3.9: Average input speed of the technical evaluation of EyePassShapes. All combinations performed equally well.

With respect to the available eye tracker, the final and rather simple question was on the size of the grid. Since the eye tracker used in the experiments works position based, it was important to choose an appropriate pixel value (a grid size) as a border value that denotes whether a stroke has been performed or not. With respect to the screen size and resolution of the eye tracker, this value has to be chosen carefully. Due to informal evaluations and analyses we favored values of 100 or 150 pixel.

The details of the technical evaluation can be found in [27]. Only the results will be quickly summarized here. Four different configurations have been validated against each other: dots + 100px, dots + 150px, grid + 100px and grid + 150px. The background pictures were optimized for the respective pixel size. The different settings were evaluated in randomized order with ten participants. Table 3.1 and figure 3.9 show the error rates and average authentication times for the different settings. Error rates and interaction speed were both slightly higher for the grid background. Even though these results have a tendency to support the dotted background, a 2 x 2 (*background image* x *grid size*) within participants analysis of variance that was performed for error rate as well as input speed showed no significant main effects and no interaction effects (all p>.05).

However, there was a subjective tendency to prefer the dotted background. Additionally, the analysis of the questionnaire showed that with six out of ten, slightly more participants preferred using the dotted to using the grid background. These factors motivated our decision to use the configuration "dotted 150px" for the main evaluation of the EyePassShapes prototype.

Theoretical Security Analysis

The theoretical security of EyePassShapes is the same as for the EyePIN system. Camera based attacks are much harder to perform and shoulder surfing is practically impossible. Using PassShapes as the authentication token, the password space as well as the possibility for dictionary attacks and the like is different. Educated guessing attacks as well as dictionary attacks are theoretically harder depending on what kind of gestures users will choose. It can be expected that as for passwords and PINs, users will be more likely to choose simpler shapes like triangles or squares. This would enable theoretical attacks as known from PINs. Educated guessing attacks however are not as strongly influenced by “bad choices” since it is harder to express names, birthdays etc. in a shape.

The password space and therefore brute-force attack resistance is strongly dependent on the length of the PassShapes and their vertical and horizontal extension. For instance, the theoretical password space for the shapes used in this study is 17,473, based on five-stroke PassShapes with a horizontal span of three and a vertical span of two. This is almost twice as big as for a four-digit PIN. A detailed password space analysis of PassShapes can be found in [132].

Usability and Security Evaluation

The prototype, based on the technical evaluation, was used for a thorough usability evaluation of EyePassShapes. The setting of the study is shown in figure 3.10. The whole process was recorded with two cameras, the first one positioned directly opposite of the participant, filming the face. The second camera filmed the keyboard respectively the touchpad. To monitor and control the study, an additional screen and keyboard was set up. The video material was used for the usability as well as the security analysis.

EyePassShapes was compared to three authentication systems: standard PIN, PassShapes [132] using a tablet PC with a touchpad, and EyePIN. EyePIN and EyePassShapes were installed on the eye tracker, PIN and PassShapes on the tablet PC. The interaction with the tablet PC was done with a pen.

User Study Design A *repeated measures within participants factorial design* was used for the study. The independent variable was *authentication method* with the levels PIN, PassShapes, EyePIN and EyePassShapes. Thus, four different authentication mechanisms were compared to each other. Standard PIN-entry represents the control condition, the baseline to judge the performance of the other systems. The dependent variables measured were error rate, speed, user satisfaction and practical security.

Procedure and Participants The study started with a detailed explanation of the different systems and tasks. After drawing an ID from a bowl, the questionnaire was handed out to the participants. The first two pages, collecting demographic data as well as eye tracking and touch

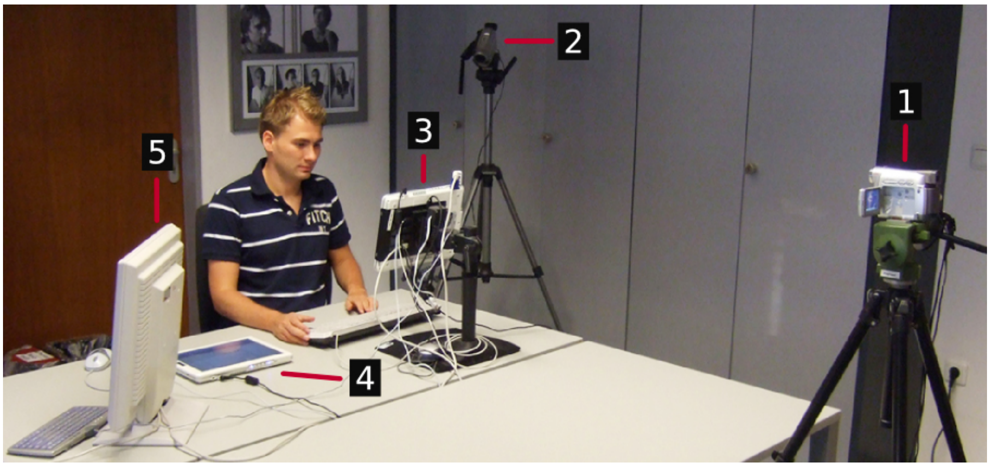


Figure 3.10: EyePassShapes user study setting: 1. Front camera. 2. Back camera. 3. Eye tracker. 4. Tablet PC for PassShapes and PIN. 5. Surveillance monitor.

pad experience, had to be filled out immediately. The rest of the questionnaire contained questions about the different prototypes and had to be answered after the respective systems had been tested.

For each prototype, the participants were provided with a thorough introduction followed by a trial phase that ran until the users felt familiar with the system. For the training phase, the participants could either choose their own authentication token or use one of the randomly provided tokens. When the participants felt ready, they were asked to draw a random four-digit PIN or five-stroke PassShape from another bowl, depending on the system. As mentioned before, five-stroke shapes have a password space almost twice as big as a four-digit PIN. However, they are the closest possible approximation. For each system, a new authentication token was drawn to minimize learning effects based on familiarity with the PIN/PassShape. For each token, the participants had three tries to authenticate themselves. After a successful authentication attempt or if failed for three times, the next part of the questionnaire was handed out to the participants before switching to the next system. Each part of the questionnaire contained questions about ease-of-use, speed and security of the respective system. Finally, the last part of the questionnaire was given to the participants asking them to rate the systems with respect to each other. For EyePassShapes, the participants could again decide themselves whether to perform the PassShape in one time or in several consecutive attempts by repeatedly pressing the trigger button.

The security analysis of EyePassShapes was based on the recorded video material. In this analysis, the term *security* refers to whether the authentication tokens (PINs and PassShapes) of the different systems can be stolen via visual attacks like shoulder surfing or video recording. We wanted to find out whether the recorded information is sufficient for an attacker to extract the correct PIN or PassShape. EyePassShapes is fully resistant to shoulder surfing attacks. That is, an attacker cannot steal the password by simply standing close to the person using the system. That is why in this analysis, we employed a highly advanced attack based on video recordings.

For the study, 24 volunteers with an average age of 28 years were recruited. The youngest one was 22 and the oldest one was 40. 16 of them were male, eight were female. The majority (19 out of 24) had never used an eye tracking system before. Only 8.3% stated that they had never used a touchpad before. Using 24 participants allowed perfectly counterbalancing the four authentication systems to minimize learning effects.

Hypotheses Keeping the results from the technical evaluation and the preliminary study of EyePIN in mind, the following main hypotheses were stated. EyePassShapes is:

(H1) easier to use than EyePIN.

(H2) faster than EyePIN.

(H3) slower than standard PIN-entry.

(H4) more secure than standard PIN-entry.

(H5) more secure than PassShapes.

Results

Authentication Speed The recorded authentication times are based on detailed log files. Each event like key presses, strokes etc. was logged, together with a timestamp. For this evaluation, the decision was made to compare pure authentication times. That is, no additional times like the one needed for pressing the “ok” button were added. Times were measured the following way: PIN was measured from the pressing the first digit to the last. The times for PassShapes were measured from the first contact of the pen with the touchpad surface till the pen was lifted for the last time. EyePIN and EyePassShapes measurement was done from pressing the control key for the first time to releasing it the last time.

Figure 3.11, left, outlines the results for the different methods. Standard PIN-entry was the fastest method (M=1.9s, SD=1.0s), EyePIN was by far the slowest input method (M=48.6s, SD=36.7s). Surprisingly, EyePassShapes performed rather bad (M=12.5s, SD=16.6s) even though we expected it to perform similar to PassShapes (M=5.8s, SD=2.1s). This was even more surprising since EyePassShapes performed noticeably better during the technical evaluation.

A one-way repeated measures analysis of variance showed that the authentication method had a highly significant influence on the input speed ($F_{1.34,28.17}$=25.14, p<.001). A post hoc analysis revealed that standard PIN was in a significant way faster than PassShapes and EyePIN (both p<.001) and was also significantly faster than EyePassShapes (p<.05). This result supports hypothesis (H3). The advantage of EyePassShapes compared to EyePIN was significant as well (p<.05), which supports hypothesis (H2). All other differences between the input methods were highly significant (all p<.001) with one exception: no significant differences could be found between EyePassShapes and PassShapes (p>.4).

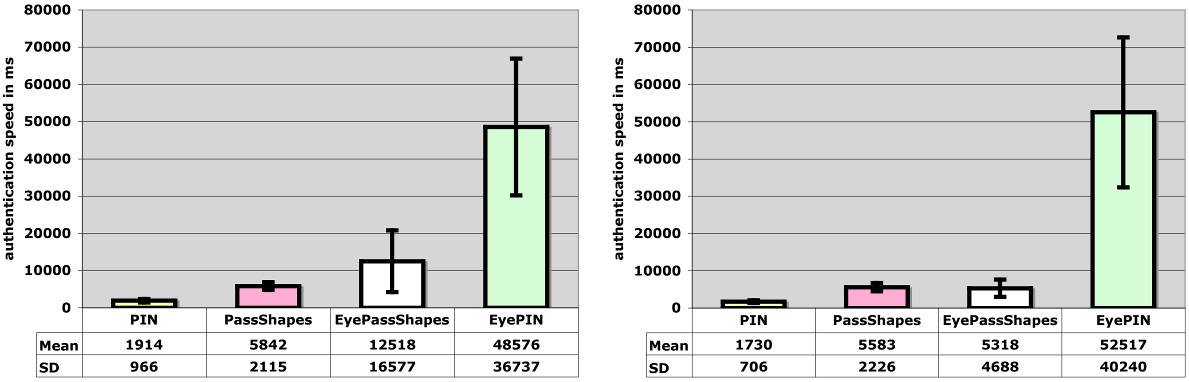


Figure 3.11: Average authentication speed in milliseconds for the different authentication mechanisms in the EyePassShapes study. Left: Data of all users. Right: Data of the users that performed EyePassShapes in one single stroke.

The surprisingly low input speed of EyePassShapes combined with the non-significant result of the comparison between PassShapes and EyePassShapes required to a deeper analysis of the data for further clarification. It shows that the difference in input times between PassShapes and EyePassShapes happened due to a group of six participants that did not perform EyePassShapes authentication in one stroke but in several consecutive strokes. That is, in contrast to the technical evaluation, some participants of the usability study decided to use the accumulative input technique for EyePassShapes.

We conducted an additional analysis splitting the results into two groups: one for the participants who had performed EyePassShapes in one stroke and one for those who used the accumulative method. The results showed that EyePassShapes was way faster when performed in one stroke (M=5.3s, SD=4.7s) than using the accumulative method (M=31.7s, SD=21.9s). When this insight was taken into account – i.e. considering only the data of the participants that performed EyePassShapes in one stroke – the results show a different picture as outlined in figure 3.11, right. A one-way repeated measures analysis of variance on the data set showed similar results when compared to the analysis of the whole data set. The authentication method highly significantly affected the input speed ($F_{1.03,15.43}$=18.85, p=.001). Standard PIN-entry was significantly faster than the other methods (all p<.05, some p<.01). EyePassShapes being faster than EyePIN was significant as well (p<.05). Those results give further support for (H2) and (H3). The small difference between PassShapes and EyePassShapes was not significant.

These results mostly match the subjective opinion of the participants. In the questionnaire, they were asked to rank the authentication methods regarding their speed. On average, standard PIN ranked first (M=1.04), PassShapes second (M=1.96), EyePassShapes (M=3.0) third and EyePIN (M=3.5) fourth.

Error Rate To decide upon the practical value of an authentication mechanism, the error rate is an important indicator as well. Since for most public terminals, authentication attempts are limited to three tries – otherwise the bank card, credit card or access right might become blocked permanently – the error rate is crucial. Based on the definition of errors introduced in chapter 2.2, for this evaluation critical errors only were considered, meaning that a participant could not correctly authenticate to the system within three tries.

To our surprise, overall only two critical errors occurred, both with EyePassShapes. In chapter 3.2.2, the high error resistance of EyePIN was already explained based on the fact that a gesture for a digit is either recognized or not and it is very unlikely to input a wrong digit. Even though the results of the comparison between EyePassShapes and EyePIN are not significant, it can be argued that EyePIN has an advantage regarding the error rate.

User Satisfaction The analysis of authentication speed gave first indications on the ease-of-use of the different systems. Further insights are based on the subjective opinion of the participants. Firstly, the questionnaire contained questions in which the users were asked to rate the ease-of-use of the different methods on Likert scales from 1 (very difficult) to 5 (very easy). Additionally, the users were asked to rank the different systems with respect to each other (ranks from 1 to 4). Another question that could give hints on the ease-of-use was on the experienced stress when using the different methods.

The evaluation of the questionnaire showed that standard PIN was rated the easiest (M=4.96), followed by PassShapes (M=4.13). EyePassShapes (M=2.67) and EyePIN (M=2.25) were rated averagely difficult. The fact that 19 of the participants had never used an eye tracker before but most of them were familiar with touchpads may have influenced those results.

Regarding ease-of-use, PIN was ranked first (M=1.13) and PassShapes was second (M=1.88). EyePassShapes (M=3.25) and EyePIN (M=3.29) ranked almost equal. The same number of participants ranked EyePassShapes better then EyePIN and vice versa. This is somewhat surprising since the results of the interaction speed analysis showed that EyePassShapes was significantly faster than EyePIN. The results of the question regarding experienced stress were highly consistent with those results. Thus, hypothesis (H1) can only be conditionally accepted.

Security As mentioned before, each participant was filmed from the front and the side while using the different authentication mechanisms. Figure 3.10 shows the position of the cameras, the respective perspectives are depicted in figure 3.12, left. That is, for each system 24 attempts were recorded from two angles. For the security analysis, only successful authentication attempts were taken into account. That means that for EyePassShapes, only 22 attempts were used.

In preparation for the security analysis, the video material was preprocessed, cut and ordered. To simulate the most effective attack possible, the final videos started when the authentication started and ended the moment the last number or stroke was input. Most effective means that the attacker did not only have the recorded material but also the information about the exact timing (when the control key is pressed the first time and released the last time). This is important since gestures also occur in normal gaze [43] and thus knowing the point in time when the authentication started is a serious advantage for the attacker. Any additional information within the videos that could

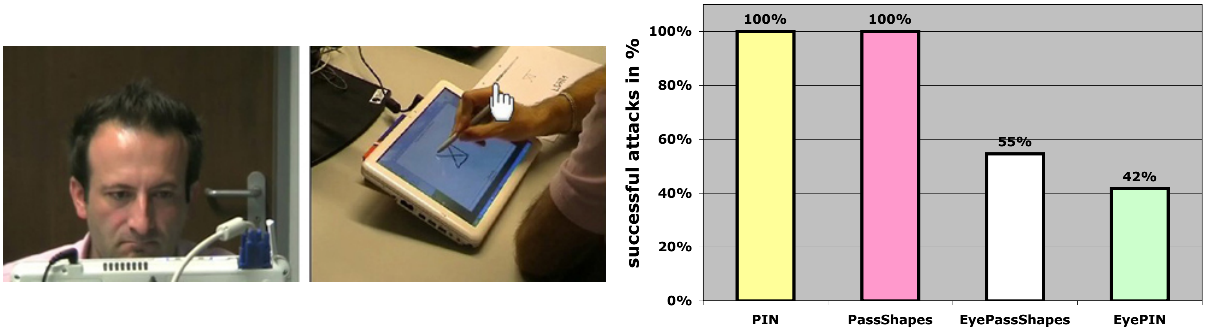


Figure 3.12: Left: The video material used for the security analysis. A front camera was filming the user's face and a back camera was used to film the hands. Right: Percentage of successful attacks on the different authentication methods.

reveal the PIN or PassShape was made unrecognizable. For instance in figure 3.10, left, the random PassShape of the user was visible on a paper and has been hidden with a hand icon.

The attacker had neither been present during the user study nor participated in the creation of random PINs and PassShapes for the study. Thus, no helpful background information was available to that person. However, that person was an expert on EyePassShapes, PassShapes and EyePIN and thus had the best qualification for an attacker. To analyze the video material, the attacker was allowed to use any video player and watch the clips as often as required. Additionally, the attacker made notes on a list. During the analysis, a second person (the observer) was present who had a list with the correct PINs and PassShapes. Whenever the attacker wanted to give a guess, the observer replied with a simple "correct" or "wrong" statement. Whenever the attacker guessed correctly within three tries – which is the standard number of trials for ATMs – the PIN or PassShape was marked as recognized.

Figure 3.12, right, shows the results of the security analysis. Due to the near to perfect observation of the input, all PINs of the standard PIN-entry and all PassShapes of touchscreen based PassShapes could be identified. The rates for EyePassShapes (55%) and EyePIN (42%) are around half that rate.

A closer look at the results reveals an interesting trend. While almost all PassShapes and PINs could be identified in the first try, for EyePassShapes and EyePIN partially the second or third try was necessary for a successful attack. This can be explained by the fact that often strokes appear similar to each other. For instance, a stroke up ("U") can easily be mistaken for a stroke up to the left ("7"), which then could be corrected in the second or third try.

A one-way repeated measures analysis of variance showed that the security of the authentication process was highly significantly affected by the system used ($F_{3,63}$=18.56, p<.001). Post hoc tests revealed that the difference in successful attacks of EyePassShapes compared to standard PIN and PassShapes was significant (both p<.05). These results support hypotheses (H4) and (H5). The difference between EyePIN and PIN respectively PassShapes was highly significant (both p<.001). No significance could be found for the differences between EyePIN and EyePassShapes.

These positive results for EyePIN and EyePassShapes are supported by the questionnaire in which the users were asked to rank the different authentication system with respect to their security. For each system, a rank from 1 (first, best) to 4 (last, worst) should be given. On average, EyePassShapes was considered the most secure system (M=1.75) closely followed by EyePIN (1.96). PassShapes ranked third (M=2.67) and standard PIN was ranked the least secure (M=2.92).

Additionally, we conducted an analysis whether the choice of the input strategy for EyePassShapes had an influence on the recognition rate. As mentioned before, EyePassShapes enables the users to input their PassShapes as one stroke or in several consecutive strokes. Out of the 22 successful authentication attempts using EyePassShapes, 16 chose the one time strategy and six the accumulative method. Only one out of the six (17%) attempts using the accumulative method could be identified while eleven out of 16 (69%) attempts performing the PassShape in one attempt were found. It has to be noted again that this result is supported by the fact that the attacker knew exactly when the gestures started and when they ended.

Due to the small group of participants using the accumulative method, this large-looking difference is not significant (p>.05). However, it supports the assumption that PassShapes performed in one gesture are less secure despite being faster. More generally, one can say that the main challenge of successfully stealing a PassShape lies in separating willingly performed eye movements from naturally occurring ones.

Discussion

EyePIN and EyePassShapes were both thoroughly evaluated in several studies. An additional memorability study, which is described in [27] was conducted as a follow up to the PassShapes memorability study reported in [132]. It confirmed the results found in the first study and attested EyePassShapes good memorability properties especially when a repeated input strategy is used. Keeping this in mind, the overall results for EyePassShapes are highly encouraging.

Standard eye tracking input methods rely on accurate positioning and thus the users need to be tracked by the system which then can be calibrated. This requires expensive hardware. EyePassShapes and EyePIN do only require data about relative eye movements. Therefore, low weight and low cost eye trackers are fully satisfying. Such an eye tracker could be realized by adding a miniature camera to the public terminal which is rather cheap or using the cameras that public terminals sometimes are already equipped with. Security-wise, the same technology could be used by an attacker. Fortunately, as for EyePIN, timing issues of key presses are essential for a successful attack, which means that a successful attack requires a second camera or another sensor.

The question whether EyePassShapes is appropriate in terms of time-critical tasks is hard to answer but at the same time very interesting. The results indicate that this question is directly connected to the question about the security of the system. Firstly, EyePassShapes can be performed very fast. The results of the usability analysis showed that it is as fast as PassShapes or better. However, this is only the case if the one stroke approach is used. That is, the whole shape is performed as one gesture and not accumulatively. Using the accumulative method takes around six times longer on average. On the other side, the results of the security evaluation revealed that it is easier for an attacker to steal the PassShape of a user when EyePassShapes is performed in one stroke. Nevertheless, both approaches are significantly more secure than standard PIN or standard PassShapes on a touchpad.

This produces a dilemma: EyePassShapes can either be used the fastest or most secure way possible. A possible solution for this problem lies in providing this fact to the users of the system. In a crowded and busy situation, they could choose to use the faster method while in a quiet and more relaxed situation, the more secure method could be applied. One way or the other, EyePassShapes is 100% resistant to shoulder surfing attacks.

EyePassShapes has the potential to fix the main problems of hardware based authentication mechanisms and could solve the performance issues of EyePIN while keeping its advantages. An interesting aspect is to find out whether users would actually change their input strategy. That is, if they would use the faster one-stroke approach in crowded situations and the more secure accumulative method without time pressure. An aspect that deserves more attention in future work is on memorability issues if users have to remember and regularly use several authentication tokens, which can be a major issue for new authentication systems [48]. Based on the fact that EyePassShapes requires a camera – even though a cheap, low resolution camera is fine – it is especially fitting for scenarios where cameras are already present. This is the case for most ATMs, having cameras attached to film the users' faces for security reasons.

3.3 User Owned Device Based Approaches

Work on authentication mechanisms for public terminals based on user owned devices is still rather rare. There are several reasons for this, the most prominent one being that the wide deployment of appropriate hardware, like smartphones has just recently started. Even though it has its very own special issues that come with connection and mostly using a wireless channel, this field has a lot of potential. Therefore, in this thesis, two authentication mechanisms that rely on the user's mobile phone were developed and evaluated. These explorations helped to reveal some interesting issues, not only about how connection (time) influences performance and acceptance but also on behavioral factors that influenced the design of the criteria.

3.3.1 1+1 = 1! Connection Issues

An essential aspect of systems based on user owned devices is how they communicate with the terminal. Simply said, it is important to make the terminal and the mobile device act as a single unit so that authentication itself can start. Considering the fact that authentication is seen as a necessary evil that is never the users' primary task [133], we have to deal with a situation in which another evil even comes before that evil. This connection dilemma comes with two major issues.

The most noticeable problem is overhead. Connection is a task that, depending on the method, takes a lot extra time. Even though techniques and technologies improve, it is still not a minor task – and will not be in the near future even though some technologies like NFC seem to have the potential to solve this problem (see for instance [101]). For the users, this means that before the active authentication can start, they have to perform a cumbersome task beforehand. That is, in many scenarios, connecting a mobile device to a terminal might require too much overhead which consequently disqualifies the method. On the other hand, for some scenarios that require a connected mobile device anyways, this overhead does not play a major role.

Another issue is security. Both, during establishing the connection and once it is connected, the communication can be victim to man-in-the-middle attacks and other attacks on the communication channel. That is, the connection technique does not only have to be fast, it has to be highly secure as well. Line based instead of wireless communication could solve both problems but at the same time would open new security holes (e.g. manipulations and camera attacks), thus nihilating the advantages of user owned hardware based approaches.

Establishing secure connections in public spaces is a big research field and discussing it thoroughly is out of scope of this work. However, the approaches as used within this thesis will be shortly outlined. Since both systems use mobile phones, and cameras are a standard equipment for these devices nowadays, the decision was made to rely on a visual marker-based connection mechanism similar to the one presented by Claycomb et al. [19, 20]. To connect to a terminal, the user makes a photo of a visual marker displayed on the terminal's screen. It contains all the information necessary to establish a secure connection. Using this visual channel, the technique is highly secure. At the same time it is very simple and intuitive since making a photo is something that is both easy to do and easy to communicate to a user. Additionally, it is very simple to deploy since the only hardware required at the terminal is a (low resolution) screen. The next chapters will describe how this connection performed in a real prototype and what it taught us about time issues in general.

3.3.2 MobilePIN

MobilePIN [29] follows a very simple basic assumption: If the PIN-entry device (i.e. the keypad) is dislocated from the terminal, conventional skimming attacks lose their effect[4]. In 2008, we designed and evaluated a mobile phone based system to securely input data on public terminals [28]. Later we enhanced it with automatic form filling functionality [94]. Similar scenarios had also been envisioned before, for instance by Sharp et al. [118].

Concept

MobilePIN can be seen as a keypad that is separated from the terminal. That is, the authentication token remains a standard PIN with all its pros and cons. Dislocating the PIN-entry comes with ad-

[4] This chapter is partially based on a paper that we published in 2009 [29].

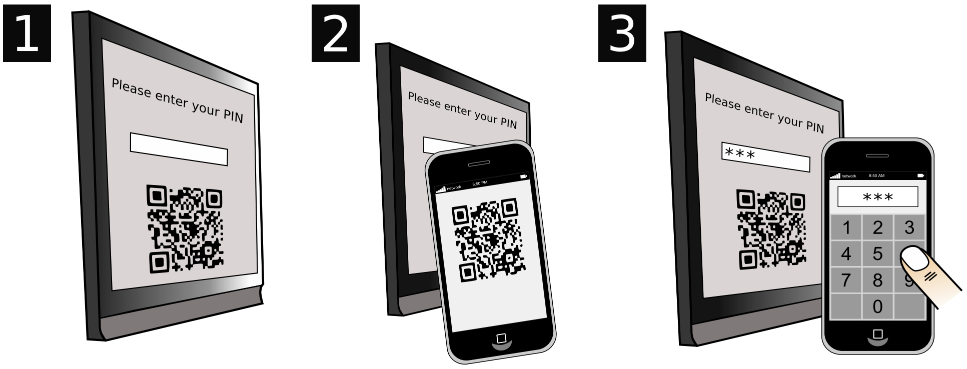


Figure 3.13: Concept of the interaction with MobilePIN: 1. A visual marker is displayed on the terminal screen. 2. To establish a secure connection, a photo is made with the mobile phone. The data on the marker is analyzed and the secure connection is established. 3.The PIN is entered on the mobile device and securely transmitted to the terminal.

ditional steps required for authentication. To authenticate on a public terminal using MobilePIN, the following steps have to be performed as shown in figure 3.13:

1. The terminal creates a visual code including the wireless address of the terminal as well as an authentication token and displays it on the screen.
2. The user takes a photo of the visual marker with the built-in camera of the mobile phone. The information on it is used to establish a secure connection between the mobile device and the terminal.
3. PIN-entry takes place at the mobile device. The digits are transmitted to the terminal using the secure channel.

For interoperability with existing terminals, MobilePIN has been designed to support standard PIN-entry and security enhanced mobile PIN-entry in parallel. This way, it is theoretically suitable for users without mobile devices and others who cannot or do not want to use mobile input. However, not using MobilePIN, the security benefits are lost. In addition to enhanced security, MobilePIN has the further advantage of enabling authentication for terminals that have none or limited input capabilities. That is, MobilePIN could be used to authenticate to any public terminal as long as they have a screen. No further input or output capabilities are required.

Prototype

To evaluate the idea of a dislocated keypad, the prototype shown in figure 3.14, left, was implemented. It has two main components, a desktop application written in JavaSE and a mobile application written in JavaME. Connection-wise, we decided to use a marker-based connection technique similar to the approach proposed by Claycomb et al. [19, 20] as mentioned before. At the beginning of each authentication attempt, a visual marker is created and displayed on the

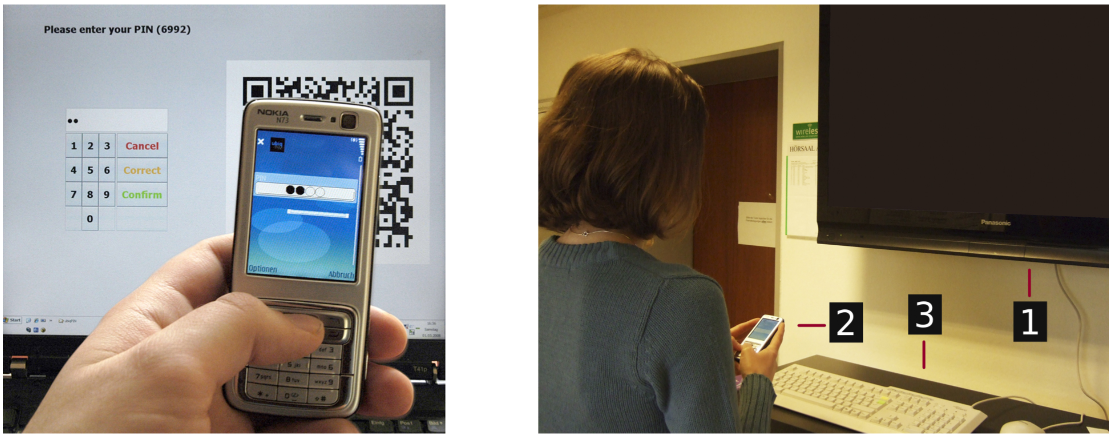


Figure 3.14: Left: The MobilePIN prototype is connected to the terminal and two digits have already been input. The output is obfuscated using asterisks both on the terminal and on the mobile phone. All input is synchronized with the terminal. Right: Setting of the user study. 1. Public terminal. 2. Mobile device the user was equipped with. 3. Keyboard for standard PIN-entry.

screen. The marker includes all the information necessary to establish a secure connection between the mobile device and the terminal. This decision was made since this technique is very easy to deploy (the terminal only needs a screen to display the marker) and it is reasonably secure. Theoretically, MobilePIN has been implemented in a way to support any connection technique, which is able to establish and provide a secure channel between the user's mobile device and the terminal.

The mobile phone application starts with a camera screen that instructs the user to make a photo of the marker in order to establish the connection. The following screen shows a simple input field that consists of four slots for the respective digits of the PIN. Input is done via the mobile phone's keypad. The only feedback given are asterisks (or better said filled black circles) on the mobile screen as well as on the terminal screen. The terminal application does not allow any input once the device is connected. However, as mentioned before, it allows input if no connection was established to improve accessibility.

Theoretical Security Analysis

Relying on PIN as an authentication token, MobilePIN has the same weaknesses as any PIN based system. That is, educated guessing, dictionary, brute force and other attacks can harm the system.

With respect to skimming or shoulder surfing, MobilePIN is resistant against most of the common attacks on public terminals. For instance, cameras directed at the keypad are of no use, since they work based on the assumption that the input device is always located in the same place and can be filmed. Users can render camera attacks useless by moving and holding their devices in different positions or even hide them inside their bags or pockets. The fact that users have different body sizes can already be considered a security advantage in this sense.

Attacks based on manipulations of the input hardware are also useless since MobilePIN works without any physical contact to the terminal. That is, the keypad stays with the user before and after the interaction which makes manipulations extremely hard or impossible. Security can even be increased if MobilePIN is used in combination with a master password, which grants access to the stored PIN. That would mean that the PIN a user inputs is different from the one transmitted to the terminal. This would render shoulder surfing attacks useless because thieves would have to steal both, the master password and the mobile device. However, not using a master password means that the system is not completely secure against a clever shoulder surfer.

Since MobilePIN requires a wireless connection to the terminal, sniffing or man-in-the-middle attacks have to be considered. Therefore, the connection algorithm has to be chosen carefully with respect to security.

Evaluation

The study took place in a laboratory at our premises. No other people could access that space during the experiment. The study setting is shown in figure 3.14, right. A laptop computer with a standard commercial keyboard attached to it was used to simulate the terminal. They keyboard was connected to a large public display. To avoid influences of the users' mobile phones on the experiment, we decided to let every participant use the very same mobile phone, a Nokia N73. The prototype software was installed on the phone and ready to use. In addition to MobilePIN, a standard PIN-entry system using the keyboard was implemented. This software was used as the control condition of the experiment.

User Study Design For the MobilePIN user study, we used a *repeated measures within participant factorial design*. The independent variable was *technique* with two levels: MobilePIN and standard PIN-entry, which was used as the control condition to judge the performance of MobilePIN. The dependent variables measured were input speed, error rate, user satisfaction and experienced privacy/security. The order of technique was counterbalanced between the participants to minimize learning and ordering effects. There was no practical security evaluation since skimming attacks would have been useless anyways and a shoulder surfer seemed not appropriate in the lab setting and thus would have most probably led to useless results. However, it can be assumed that a clever shoulder surfer would be able to steal nearly all authentication tokens if no master password was used.

Procedure and Participants For each participant, the exact same procedure was applied. They were brought into the room where the tasks were explained to them in detail. To keep the experiment as unbiased as possible, the explanation was written down and was read in exactly the same way to each participant. After that, the participants were asked to draw a random number from a bowl which was used for anonymous identification and for connecting the log files (including the measured times, errors etc.) to specific participants (respectively the questionnaire).

Each participant performed two different tasks. Each task was to enter a PIN correctly, with the standard keyboard (task 1) and with MobilePIN (task 2). The PIN was displayed to the users on the screen of the terminal and was randomly generated. For each task, a new PIN was assigned to the user. Counterbalancing was achieved by assigning the order of the tasks to a participant with respect to the number drawn from the bowl. Odd numbers started with task 1 while even numbers started with task 2. For each task, the users had three tries to correctly authenticate to the system. If they authenticated correctly or failed for three consecutive times, the task ended. After finishing both tasks, each participant was asked to fill out a questionnaire. This was done to collect information about user preferences as well as basic demographic data. Likert scales from 1 (do not agree) to 5 (highly agree) were used in the questionnaire. Additionally, all interaction was logged with timestamps by the system for later analysis and evaluation.

The user study was conducted with 19 volunteers, thus no perfect counterbalance was achieved. However an imbalance of one seemed to be acceptable. The average age was 25 years and the male/female ratio was almost 50/50 with 9 female and 10 male participants. The youngest participant was 20 years old, the oldest 32 years. Asked about how many times they withdraw money from an ATM per month, the average answer was 4.6 times, while 1 was the smallest and 15 the highest value.

Hypotheses Based on the theoretical analysis of MobilePIN and our experience with previous systems, the hypotheses were the following:

(H1) PIN-entry with MobilePIN is slower than with the keyboard of the terminal.

(H2) PIN-entry with MobilePIN is more error-prone than standard PIN-entry.

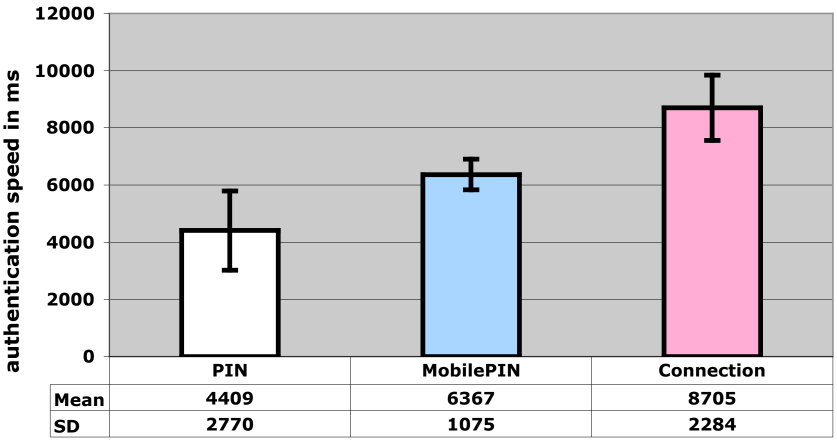
(H3) Users will consider MobilePIN more secure than standard PIN-entry.

Results

Authentication Speed To measure input speed, the time between the first press of a button and the correct PIN being confirmed was analyzed. Out of the 19 samples, two extreme outliers had to be removed for the analysis.

Figure 3.15 shows the mean times and standard deviations of the different systems and the connection times. A paired-samples t-test was used to analyze the data. It showed that the time needed to input the correct PIN using MobilePIN (M=6.4s, SE=0.26s) was significantly slower than using standard PIN-entry (M=4.4s, SE=0.67s, *t*(16)=.2.92, *p*<.05, r=.59), which supports hypothesis (H1). One interesting finding is that even though the average times for standard PIN-entry are significantly lower, there has been a higher diversity in results than for Mobile PIN.

The results show that users are faster with standard keyboards than using a mobile phone. More interesting is the overall time of the interaction. While this time was the same for standard



	PIN	MobilePIN	Connection
Mean	4409	6367	8705
SD	2770	1075	2284

Figure 3.15: Average authentication speed of the MobilePIN study. Time is measured for the plain input plus confirmation only. Additionally, the graph show the average connection times that add significant overhead to MobilePIN.

PIN, MobilePIN has an additional overhead created by the technology used for the connection between the mobile device and the terminal. The marker-based approach chosen for MobilePIN created an average overhead of 8.7 seconds. This is more than the average time for entering a PIN using MobilePIN. Thus, it creates an overhead of more than 100%. This shows that, despite its many advantages, the main problem of the marker-based approach is speed. Even though the participants rated the speed as ok, we argue that a faster connection technology is likely to increase acceptance of a system like MobilePIN.

Using a public terminal or service that requires a connected mobile device (e.g. [28]), connection time can be neglected. That is, in some cases, connection can be considered as part of the main service and thus not part of the authentication mechanism. However, such services are still rare, even in research and thus, in the current situation, it is fairer to consider connection establishment as part of the authentication mechanism.

Error Rate A comparison of error rates for the two systems showed a surprising result. Again we distinguished between critical and non-critical errors. Critical errors are defined as three wrong tries, non-critical as one or two failed attempts. Surprisingly, neither standard PIN nor MobilePIN resulted in any critical errors and only one non-critical error occurred for each of the systems. Besides errors, the system additionally logged corrections. A correction was performed whenever the users deleted the input to restart PIN-entry. The participants were told that they could correct their input as many times as they wished and should do so if required. Again, results show very low correction rates. While only one correction had been applied for standard PIN, MobilePIN needed no correction at all. Based on these results, (H2) had to be rejected. We believe that this is due to increased habituation to entering information on mobile devices, especially for people within the age group of the participants.

User Satisfaction To gain a general understanding of the users' needs when authenticating with public terminals, they were asked to rate three major aspects for this kind of interaction: security, speed and error-resistance. The participants had to rate the importance of these three aspects on a Likert scale from 1 ("not important") to 5 ("very important"). The results show that error-resistance (4.5) and security (4.6) are rated notably higher than speed (3.7). This indicates that users are willing to accept slower interaction methods if they increase security. Noticeably higher error rates seem rather unacceptable.

An obvious reason for the high ranking of security is the possible loss of the users' possessions in case of fraud. That is why 90% of the users claimed to use extra safety precautions when authenticating on a public terminal. Almost 50% even claim to use several extra security measures like hiding the PIN-entry with the other hand. We will see in chapter 5 that such high numbers have to be taken with a pinch of salt since field study results reveal that security precautions are rather seldom in real ATM use.

Additional questions were on subjective opinions about security of the different systems. A Likert Scale from 1 ("I do not agree") to 5 ("I highly agree") was used. The main purpose was to determine whether users felt more secure when using MobilePIN compared to standard PIN-entry. The statement "MobilePIN provides the highest possible security when entering private information" was rated 4.1 by the participants compared to 2.6 when asking the same question for standard PIN-entry, which gives support to hypothesis (H3).

Ease-of-use of MobilePIN (4.2) and standard PIN (4.7) was rated on the same Likert scale. When asked about the experienced interaction speed, standard PIN averaged 4.3 and MobilePIN 3.3 points. The lower score for MobilePIN correlates with the results from the log files. Still, the result is surprisingly good considering the huge overhead created by the connection mechanism. One possible explanation is that people enjoyed playing with the marker-based system, which they had never used before. We argue that after getting familiar with the technique, connection time is likely to be experienced as a bigger drawback.

Discussion

MobilePIN was the first exploration of user owned device based approaches within this thesis. Besides the fact that it can increase security and by relying on PIN and a device that users can handle well (mobile phone) it is very intuitive, the main lessons learned are about connection issues.

Connection time influences overall authentication speed as well as possible scenarios. Choosing a marker-based approach, a very intuitive way of connecting a mobile device to a terminal was selected. At the same time, the built-in hardware of the mobile phone was used. However, the results showed that connection time alone took more of the overall time than the active authentication itself. Thinking about a scenario in which connection is only done to make secure authentication possible, this means adding an overhead of more than 100% to the overall interaction time. Thus, not counting this time to the authentication process would be unfair. Things look different if the scenario or service relies on a connected mobile device itself. For instance, in 2008, we developed a secure input mechanism for public terminals as described in [28]. In this system, any private information is input on the users' mobile device and transmitted to the terminal. Such a system naturally requires a connected device for the service itself. Using MobilePIN to authenticate to this service could thus be considered a good solution.

Summarized, MobilePIN was the first work that provided us with hints that standard usability evaluation might not be appropriate for authentication mechanisms and that especially correctly measuring time deserves much more attention than it received till that point. This will be further highlighted in chapter 4.

In the experiment, users seemed to accept the overhead created by the connection method. It is very likely that users are willing to accept the rather big overhead in a lab situation but probably would not accept it if used in the field. Fortunately, MobilePIN can be easily modified to work with any secure connection technique. It is even conceivable to include support for several connection mechanisms and let the users or the terminal provider decide on the required level of security and convenience. It is furthermore imaginable to provide authentication mechanisms that enable connection while queuing at the terminal or even on the way to the terminal.

Currently, MobilePIN uses standard PINs as authentication tokens. That is, it inherits its disadvantages. For instance, if users have problems remembering PINs, they are likely to choose simple PINs like their birthday or write them down, which decreases security [2]. It would be interesting to see how alternative authentication tokens (e.g. graphical passwords) perform with MobilePIN. Another field that seems worth to be investigated is whether MobilePIN could provide a unified authentication method for public terminals. With this system, authentication functionality could even be added to public terminals that do not support input and thus cannot support authentication (e.g. proactive displays).

3.3.3 VibraPass

MobilePIN in its original version does not provide high security against shoulder surfers. VibraPass [34] tries to fill this gap but in a way keeping the security-related advantages of user owned device based approaches. This is done by using a technique well known from related work on hardware based approaches: using the vibration functionality of mobile phones as an invisible channel from the terminal to the user[5].

Concept

To enable secure authentication on public terminals, VibraPass introduces a concept called "lie overhead". PINs and Passwords are enriched with "lies", redundant information that does not contribute to the actual authentication token and confuses observers. The lie overhead indicates how much overhead, wrong information, is added to the input (in percentage). This additional information is randomly mixed with the real PIN or password. The knowledge about truth and

[5] This chapter is partially based on a paper about VibraPass that we published in 2009 [34].

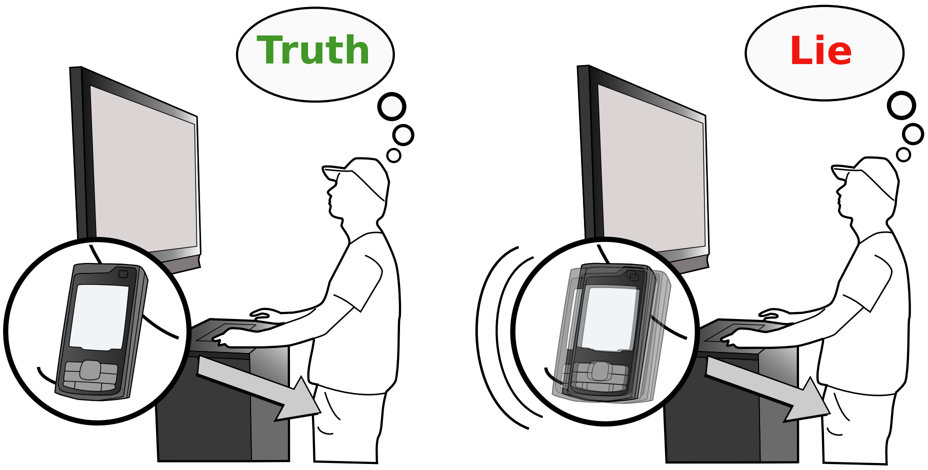


Figure 3.16: The concept of VibraPass. Left: If the mobile phone does not give any feedback, the user enters the next correct digit of the PIN. Right: If the mobile phone vibrates, the user enters a false digit.

lies is shared secretly between the terminal and the user. Therefore, the terminal, as opposed to an attacker, can extract the real authentication token from the input.

This knowledge is shared utilizing the users' mobile devices or better said their mobile phones. Each current mobile phone or PDA is equipped with vibration motors. They are used to provide haptic feedback to the users. That is, mobile phones already provide a tactile output channel, which is an appropriate way to transport simple binary messages like "true" or "false".

VibraPass works as follows:

1. The user connects the mobile phones to the terminal. This is necessary each time the user wants to interact with a terminal. This could be done using visual markers, NFC or other techniques.

2. The terminal creates a randomized sequence of lies (the lie chain) based on a pre-defined lie overhead. The randomization prevents attacks based on knowledge of the order of lies. An example sequence of lies for a four-digit PIN could be "0,1,0,0,1,0" (0 means truth and 1 means lie/vibration).

3. The user enters the PIN or password. Every time the mobile phone vibrates, the terminal indicates to the user that the next input should be a lie (figure 3.16, right). When the device remains quiet, the next part of the real password or PIN should be input (figure 3.16, left).

VibraPass provides enhanced security for authentication in public spaces while still relying on the basic input mechanisms of the respective terminal. That is, no additional hardware (besides a communication module like Bluetooth) is necessary on the terminal side. Theoretically, VibraPass could be used to enter arbitrary amounts of information on public terminals. Due to its nature of adding overhead to the input it seems more suitable for short chunks like authentication tokens.

Prototype

VibraPass was implemented as a lightweight prototype written in JavaSE for the terminal/server side and JavaME for the mobile phone application. We used Bluetooth as the communication channel mainly since it is easy and cheap. No secure connection mechanism was implemented for the prototype since this was already evaluated in-depth during the MobilePIN study. Thus, a simple Bluetooth device list was provided out of which a VibraPass client could be selected.

The main responsibility of the server was to handle the connection to the user and change the tasks automatically. As will be described later, each participant had to perform 32 authentication sessions. To keep the experiment fast, changing the sessions (counterbalanced and randomized) was done automatically by the server, depending on the participant's ID. The server logged all interaction, errors and button presses. At the beginning of a session, the server automatically created a lie chain, a secret sequence of lies mixed with the real input. Every time a lie was expected from the user, the server sent a vibration signal to the client (with releasing the previous button). The client did not have any additional user interface. Its only purpose was to vibrate whenever the server sent the respective signal.

Theoretical Security Analysis

Due to the lie overhead and the wrong information that comes with it, shoulder surfing attacks are very hard to perform or useless. Skimming attacks, that is manipulation of the keypad, filming it and the like do not reveal the authentication token as well. Even if the input is filmed, the attacker records the wrong information and has no knowledge about how to separate it from the actual PIN or password.

Attacks on the wireless channel are not any more efficient than that. Doing the input directly on the mobile device, as is the case for MobilePIN, would require transmitting the password or PIN and would make it vulnerable to man-in-the-middle attacks. In the VibraPass system, however, no sensitive data of any kind is transmitted but only "vibrate" commands. Reading the wireless channel alone does therefore not reveal the order of lies since the signals depend on the input speed of the user and not on predefined time slots. Thus, cracking the input in one attempt would require a camera attack synchronized with a sniffing attack.

The main weakness of VibraPass is that repeated observations can lead to successful intersection attacks by analyzing the differences between inputs. The highest success rate for an attack can be assumed if the lie overhead is known by the attacker. For instance, two recordings can lead to breaking a four-digit PIN when the smallest overhead is used. In real world situations this threat is considered rather minimal since manipulated terminals are usually quickly repaired and users mostly do not interact twice with the same terminal within a short time frame.

Again, attacks that rely on guessing the authentication token have the same probability for VibraPass as for any other PIN or password based approach.

Usability and Security Evaluation

For the user study, we set up a public terminal in a corridor of our labs consisting of a 42 inch screen, a laptop and a standard commercial keyboard connected to it as outlined in figure 3.16. The laptop was not visible to the participants. Overall, the physical setup of the study was very similar to the one used for the MobilePIN evaluation (see chapter 3.3.2). Two cameras, one pointing at the keyboard from the top and the second one recording the whole interaction from the right side, were installed along with two microphones (co-located with the cameras). The recordings were used for usability as well as security analysis.

To avoid influences of the users' private mobile phones on the study results, like for the MobilePIN study, participants were equipped with a mobile phone. This time, the phone of choice was a Nokia N80 that we placed in the pockets of the participants' trousers. Synchronization between the users and the terminal was simply achieved by sending the vibration signal immediately with the release of the previous button. No participant had major issues with this approach.

Connection time was neither implemented nor measured since for this aspect, we could rely on the results of the MobilePIN study. That is, the users got a device that was already connected to the terminal. In contrast to MobilePIN, in addition to PINs, passwords were evaluated as well.

User Study Design VibraPass was evaluated using a *repeated measures within participants factorial design*. The independent variables were *PwType* (random PIN, random password, user generated PIN, user generated password), *PwLength* (4 and 8) and *LieOverhead* (0%, 30%, 50%, 100%). The lie overhead of 0% represented the control condition since it is identical to standard PIN or password entry. The dependent variables measured were error rate, authentication speed, user satisfaction and practical security.

The task was to authenticate to the public terminal using every combination of the independent variables (*PwType* x *PwLength* x *LieOverhead* = 32 authentication sessions per user). The order of *PwType* was counterbalanced between the participants, while the order of PwLength and LieOverhead was randomized to minimize learning effects.

Procedure and Participants At the beginning of each session, the prototype was explained to each participant in all its details. This was followed by a training phase, in which the participants were allowed to use the system until they felt familiar with it. When they felt ready, they were asked to define two passwords and two PINs, each with a length of four and eight. Randomized passwords and PINs were provided on printed lists since we did not expect from users to remember eight new authentication tokens. Randomized passwords were generated using a vowel as every second letter to increase readability and memorability. Some examples are shown in table 3.2. Each password/PIN was only used for one participant to avoid influence on the results. In the next step, the participants were equipped with the mobile device, which was already connected to the terminal via Bluetooth.

At the beginning of each authentication session, the terminal informed the current participant, which password to choose from the lists and created a randomized lie sequence based on the

Table 3.2: Examples of randomized passwords with a length of four and eight characters as used in the VibraPass user study.

four-letter	tufi, huti, piwo, neve
eight-letter	inodokaw, asidehib, ehavemab, upimuwef

current lie overhead. Every key press, correction, error etc. were logged. For each authentication session, there was a maximum of three tries to fill in the right authentication token. Changing to the next session took place whenever the previous authentication was successful or had failed three times. In the end, each participant was asked to fill out a questionnaire. Ratings were given using Likert scales from 1 (do not agree) to 5 (highly agree).

For the study, 24 volunteers were recruited. They had an average age of 23 years, eight of them were female. At the time of the study, all of them owned mobile phones with vibration functionality. Choosing 24 participants allowed perfect counterbalance of *PwType* to minimize learning effects. Thus, the results in this study are based on 768 authentication sessions performed by 24 participants.

Hypotheses Based on pre-studies of the VibraPass system, the following hypotheses were stated:

(H1) VibraPass is more secure to observation attacks than standard PIN or password entry.

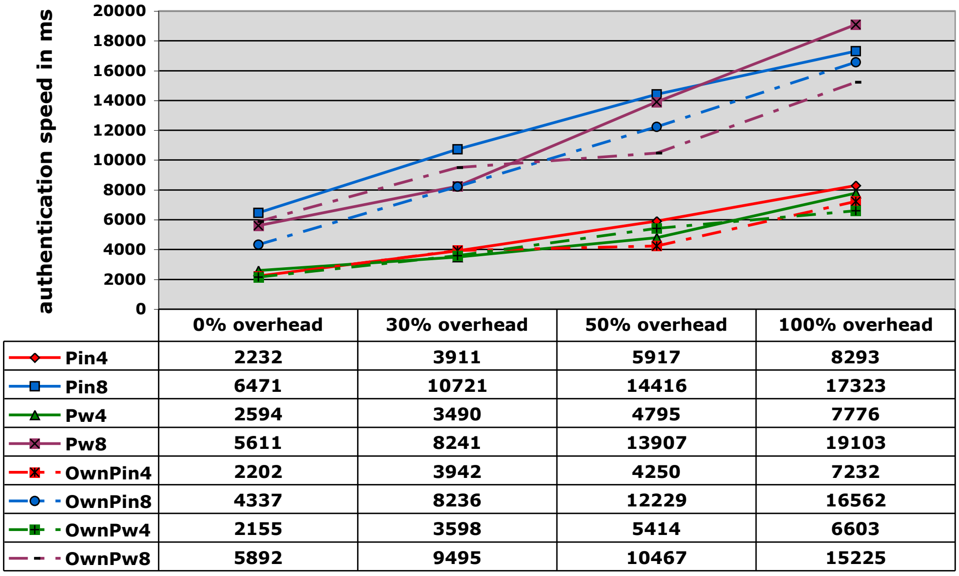
(H2) The error rate increases a) the higher the lie overhead and b) the longer the password.

(H3) Interaction time increases a) the higher the lie overhead and b) the longer the password.

Results

Authentication Speed Each user performed 32 authentication sessions. In each session, interaction time was measured from the first key press of the first digit or character to the moment when the last key was released. The measurement did not include times required to press the enter button to confirm the input. Thus, only pure active authentication time was measured. Sessions with critical errors were excluded from the analysis. Figure 3.17 depicts the average interaction times for all combinations of the independent variables. It shows that in most of the cases, interaction time increases when *LieOverhead* is increased. It also shows the increase of the time needed for authentication with PwLength of level 8 (upper four lines) compared to level 4 (lower four lines). Nevertheless, when comparing the time needed to input four-digit random PINs with lie overhead 0% (M=2.23s, SD=0.86s) to four-digit random PINs with lie overhead 30% (M=3.91s, SD=1.70s), the time needed for the more secure variant is still within a reasonable range.

A 4 x 2 x 4 (*PwType* x *PwLength* x *LieOverhead*) within participants analysis of variance showed significant main effects for PwLength ($F_{1,13}$=131.35, p<.001) and LieOverhead ($F_{3,39}$=107.59,



	0% overhead	30% overhead	50% overhead	100% overhead
Pin4	2232	3911	5917	8293
Pin8	6471	10721	14416	17323
Pw4	2594	3490	4795	7776
Pw8	5611	8241	13907	19103
OwnPin4	2202	3942	4250	7232
OwnPin8	4337	8236	12229	16562
OwnPw4	2155	3598	5414	6603
OwnPw8	5892	9495	10467	15225

Figure 3.17: Average authentication speed for the different combinations of *PwType* and *LieOverhead*. Both significantly influence speed.

p<.001). A significant interaction effect was found for *PwLength* x *LieOverhead* ($F_{3,39}$=21.06, *p*<.001). Post-hoc comparisons showed significant differences (*p*<.001) in interaction speed between *PwLength* with level 4 (M=4.65s, SE=0.33s) and level 8 (M=11.14s, SE=0.72s). Comparing the different levels of LieOverhead showed significant results as well (all *p*<.001). These results confirm hypothesis (H3) a) and b). Regarding interaction effects of *PwLength* x *LieOverhead*, the results show that changing PwLength influences interaction time when increasing LieOverhead. This result is significant for all levels of *PwLength* x *LieOverhead* (all *p*<.05, most *p*<.001).

Error Rate As for the other authentication mechanisms, two types of errors were distinguished in the analysis of VibraPass: basic errors that indicate that at maximum two tries of an authentication session failed, and critical errors, which indicate that the authentication session failed completely.

Out of the 768 authentication sessions, 63 (8%) were performed with at least one or more wrong inputs (including critical errors). Four of them (0.5%) using a LieOverhead of 0%. 19 (2.5%) sessions ended in critical errors. None with a lie overhead of 0% or 30% created a critical error. 557 out of the 576 (96.7%) sessions using a lie overhead bigger than 0% could be completed successfully.

Even though error rates for VibraPass are quite low, it is worth taking a closer look at which levels of the independent variables influence critical errors.

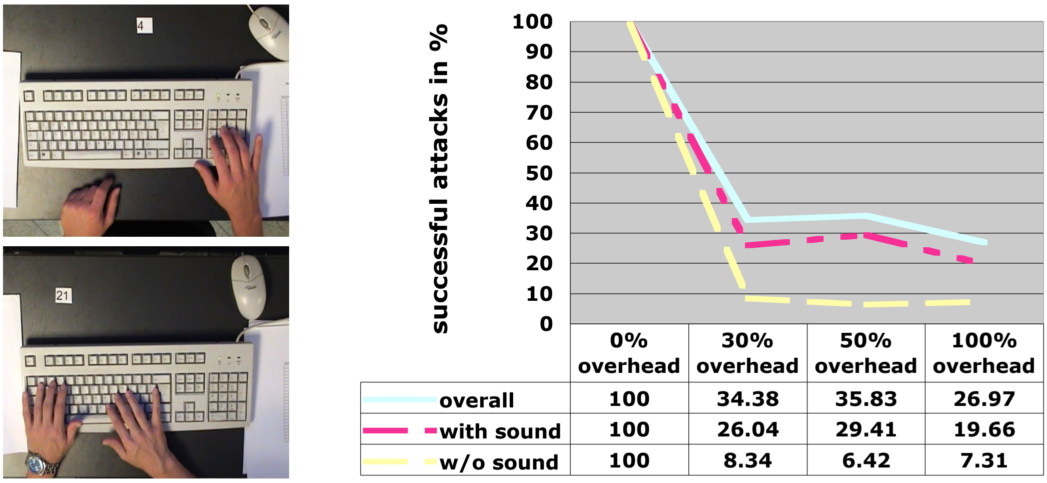
A 4 x 2 x 4 (*PwType* x *PwLength* x *LieOverhead*) within participants analysis of variance of critical errors showed significant main effects for *PwLength* ($F_{1,23}$=9.70, *p*<.05) and *LieOverhead*

($F_{3,69}$=7.24, p<.05). No significant interaction effects were found. Post-hoc tests revealed that the difference in the occurrence of critical errors using *PwLength* of 8 (15 out of the 19 critical errors) compared to 4 critical errors with *PwLength* 4 is significant (p<.05). No critical errors occurred using levels of 0% and 30%. Thus, the 19 critical errors all occurred with lie overhead of 50% (5 out of 19) and 100% (14 out of 19). The post hoc tests revealed significant differences between *LieOverhead* level 100% and 0% and between level 100% and 30%. These results mainly support hypothesis (H2) a) and b) with the exception that 30% did not lead to any critical errors.

User Satisfaction The low error rates and fast authentication speed gave first indications on whether the system might be accepted by users or not. In addition, qualitative data was collected directly from the participants. In the questionnaire, participants were asked to rate the different levels of lie overhead. The results showed that all participants preferred either a low lie overhead of 30% (13 participants) or the medium lie overhead of 50% (11 participants). Lie overhead of 0% as well as 100% were favored by none of the participants. As a reason for the likeability of 30% - 50%, most participants mentioned that they found it still very easy to use but considered it more secure. One participant called the medium lie overhead a "*good trade-off between usability and security*". The analysis of the participants' answers regarding security and ease-of-use encourage the use of a lie overhead between 30% - 50%, depending on the password length. On a Likert scale from 1 (do not agree) to 5 (highly agree), security for 30% was rated 3.6 compared to 2.3 for standard authentication. Regarding ease-of-use, both were considered fairly easy (4.7 for 0%, 4.2 for 30%, 3.2 for 50% and 2.1 for 100%).

Security The security analysis of VibraPass represented a worst-case scenario. Two video cameras were set up, one filming the keyboard from above and the second one filming the participants from the right side. Examples of the video material are shown in figure 3.18, left. The two cameras' microphones were used to record the audio part of the interaction (mainly the audible vibration). To make the attacks as efficient as possible, we assumed that the attacker knows as well the lie overhead as the length of the authentication token. Moreover, the video material was cut so the 32 different sessions per participant were easily differentiable. Finally, the used mobile phone had a rather loud vibration alert.

The video recordings were analyzed abiding to strict rules to ensure mostly unbiased results. Rules were defined based on a prior sighting of the recorded material. The main and only question was how many passwords and PINs could be stolen by an attacker. It should be highlighted that it is next to impossible to have such optimal conditions in a real world setting since public terminals are usually located in rather crowded and noisy places and that this decision was made to evaluate the security of VibraPass under most insecure conditions. The attack was performed by two persons familiar with VibraPass. They could sight the video material as many times as required and wrote down the authentication tokens, which they thought they identified. In addition to this information, they wrote down how the password or PIN was identified.



	0% overhead	30% overhead	50% overhead	100% overhead
overall	100	34.38	35.83	26.97
with sound	100	26.04	29.41	19.66
w/o sound	100	8.34	6.42	7.31

Figure 3.18: Left: Examples from the video material of the security analysis. The attacker could play the material as often as required. Right: Successful attacks on different lie overheads of VibraPass. Without sound refers to so-called "bad lies".

Out of the 749 successful authentication sessions, 100% with a lie overhead of 0% could be identified (192 sessions). No participant tried to hide any of the input which we think was due to the lab setting. However, that allows to evaluate the built-in security of the system. VibraPass enhanced authentication sessions only revealed the true password or PIN in 32.5% of the cases (181 out of 557). The main reason for successful attacks was audible vibrations, for example due to keys, coins or other items in the pockets of the participants (140 out of 181). Without such audible hints, only 41 attacks (7.3%) on VibraPass with a lie overhead greater than 0% would have been successful. A detailed breakdown of the results is shown in figure 3.18, right. A 4 x 2 x 4 (*PwType* x *PwLength* x *LieOverhead*) within participants analysis of variance showed only a significant main effect for *LieOverhead* ($F_{3,39}$=28.53, p<.001) and no interaction effects. Post-Hoc tests revealed that only the differences between *LieOverhead* 0% (100% successful attacks) and all *LieOverhead* > 0% were significant (all p<.05). These results support hypothesis (H1). We argue that in real world use, the number of successful attacks is more likely to be what we found without audible vibrations since background sounds and the public setting create noise out of which it is much harder to filter the vibration sounds.

The most interesting finding was on the reasons for successful attacks. As mentioned before, the attackers categorized how they could identify a PIN or password. Overall, seven different categories of successful attacks were identified. Exploiting audible vibrations was the security hole that had the highest success rate in identifying the correct PIN or password. There were also a few instances of repeated passwords. For instance, when users had to perform the password or PIN several times due to input errors. All other successful attacks could be summed up to what we called "bad lies".

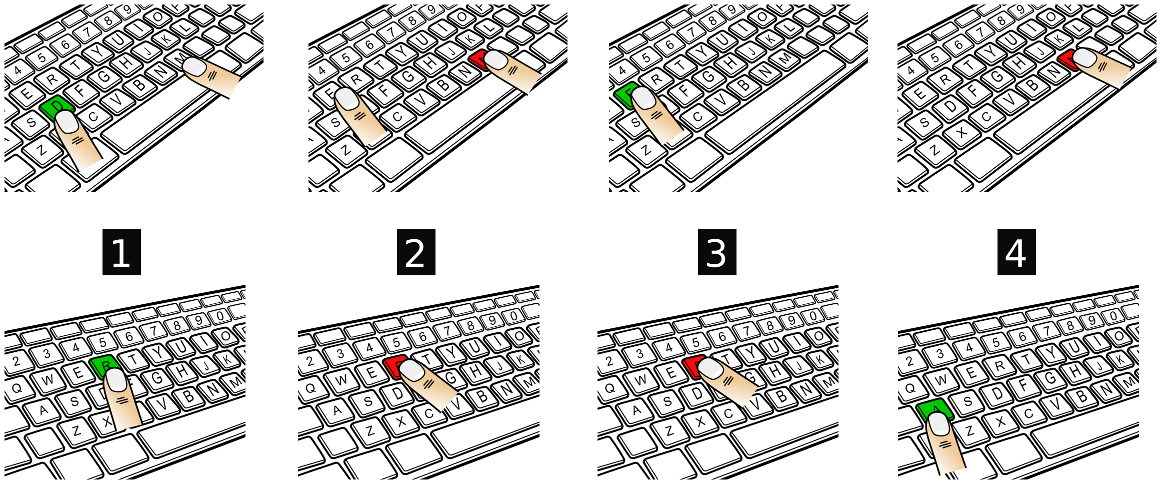


Figure 3.19: Two examples of “bad lies”. Top: In this example, the user chose the key ’M’ as a lie key that they repeatedly use (lie sequence 0,1,0,1). Bottom: The user in this example applied repeated pressing. That is, for a lie the respective preceding key is pressed (lie sequence 0,1,1,0).

Bad Lies

The security analysis of VibraPass built the foundation of the first criterion within this thesis that is not directly related to standard usability factors. As mentioned before, “bad lies” were identified as the main reasons, besides audible vibration, why VibraPass protected input could be broken by an attacker. Five main categories were identified: *repeated pressing*, *lie key*, *early finish*, *confused waiting* and “*you’re doing it wrong*”.

When a user applies *repeated pressing*, lies are done by simply pressing the key that was previously pressed to perform a real part of the PIN or password. An example is shown in figure 3.19, bottom. Such repetitions in the entered PIN give useful hints for an attacker. In the security evaluation, it led to 17 identified PINs and passwords.

Another often applied strategy was using a so-called *lie key* as shown in figure 3.19, top. In this approach, the user chooses a key for the first lie and keeps pressing it for any following lie. Often, this lie key was used with one hand, while the real digits or characters were input with the respective other hand. This revealed 15 authentication tokens in the study.

The last three “bad lies” did not lead to as many successful attacks but are still worth mentioning. *Early finish* happened when a lie was supposed to come at the end but users simply forgot to input them. *Confused waiting* refers to participants that wanted to press the next real digit or character firstly but then realized that they had to input a lie. Their finger already hovered over the next real token but then switched to another key for a lie. Finally, “*you’re doing it wrong*” happened the most seldom (only two times). This was when PINs were enriched with lies based on characters and was only possible since a standard commercial keyboard was used for both, PINs and passwords.

With less than 8% success rate, “bad lies” look like a minor problem. However, some participants kept applying the same bad strategies for several times. The most interesting aspect is that this helped us to reconsider the view on how security is provided by an authentication system. The choice that was made for VibraPass was to shift some of the responsibility for securing PIN and password entry to the user. That is, in some way, users have to behave “clever” so their data is protected. “Bad lies” impressively show that this approach is always likely to fail. Based on this, the first non-standard usability criterion within this work was defined: *security should not require an active user*. That simply means that security should be built-in to a system and not require specific “right” or “good” behavior by the user. More evidence for the importance of this criterion will be provided in chapter 5.

Discussion

The evaluation showed that VibraPass has the potential to increase security while providing low error rates and fast input speed. Especially compared to other secure authentication mechanisms, VibraPass performs extremely well with a lie overhead between 30% and 50%. It highly increases security but still provides fast input speed with low error rates. In fact, in sense of error rates, lie overhead 30% performed as good as standard password and PIN since it did not result in any failed authentication session. It also provides an input speed close to standard PIN and password entry. The qualitative data collected from the participants supports this conclusion. However, enriching the input with lies at randomized positions is very likely to have a bad influence on memorability since it destroys the muscle memory effect [117]. This might especially become an issue when several authentication tokens have to be remembered.

Even though the performance in the lab study was great, the findings on “bad lies” seem to disqualify the system as a good candidate for future use. Shifting responsibility to the user is what is already done with standard PIN-entry at public terminals. The input is only secure as long as users behave secure. This is not a big problem for VibraPass but as shown with “bad lies” is able to compromise the security of the system. We therefore argue, and define the criterion that security should not require an active user. This is the first non-standard usability criterion defined and will be outlined in more detail later in this thesis.

As opposed to other user owned device based approaches, VibraPass does not dislocate the input from the terminal while still being highly secure to skimming attacks. This makes securing the wireless channel slightly less important since the authentication token itself is not transmitted over this channel. Therefore, connection issues can theoretically be handled easier, for instance by choosing faster but less secure approaches. Nevertheless, connection time should be counted to the overall input time depending on the scenario and thus can negatively influence the good perform of the system. Highlighted once more, this means that user owned device based approaches should be especially honest on measuring time with a focus on the scenario the authentication is supposed to work in.

3.4 Software Based Approaches

Purely software based authentication mechanisms that are secure even to camera based attacks are the hardest to design. Overall, this field is therefore the most challenging. As seen in related work, even low security comes with great costs considering overhead. Especially authentication speed tends to be bad. The reason why researchers are interested in this field despite these problems is that software based approaches come with a big advantage. Deployment is extremely cheap since they rely on standard input hardware like keypads, keyboards or touch screens (that are just widely hitting the market). Usually, a software update is enough to make them work. Thinking about updating millions of terminals, this advantage cannot be played down. In this chapter, ColorPIN will be presented which has been designed with a focus on improved performance and high security, even to advanced attacks like cameras.

3.4.1 ColorPIN

Reading related work, it seems that software based authentication mechanisms are either only secure against shoulder surfing and slow or also resistant to more advanced attacks and very slow. The goal of ColorPIN [32] was to create an authentication mechanism that does not require an active user to protect the input (as learned from VibraPass). Additionally, the added overhead should be kept low and it should not require major hardware changes at the ATM. Further requirements of the system were strong resistance to shoulder surfing, camera attacks as well as other hardware manipulations. A very promising approach to achieve this seemed to be using indirect input. This means that the authentication tokens are not directly input, but instead some kind of "detour" is used[6].

Concept

When brainstorming about ColorPIN, one of the main goals was not to add overhead by requiring several inputs to enter one part of the authentication token. Therefore, to achieve higher security but remain a one-to-one relationship between the length of the authentication token and the required number of key presses, the authentication token PIN was slightly manipulated. A PIN in the ColorPIN system contains of a combination of digits, of which each digit is combined with one of three colors (black, red or white). That is, a user has to remember four colored digits and thus a four-digit PIN in the ColorPIN system could look like the following:

1 (black) - 2 (red) - 3 (white) - 4 (black)

The user interface consists of a keypad representation showing the digits 1 - 9. Digit "0" was removed due to reasons of simplicity as shown in figure 3.20. On the bottom of each number, three differently colored letters can be found. Those letters are randomly assigned at the beginning of the interaction. Additionally, due to security reasons, the letters are newly assigned each time the

[6] We published ColorPIN in 2010 [32]. This chapter is partially based on this publication.

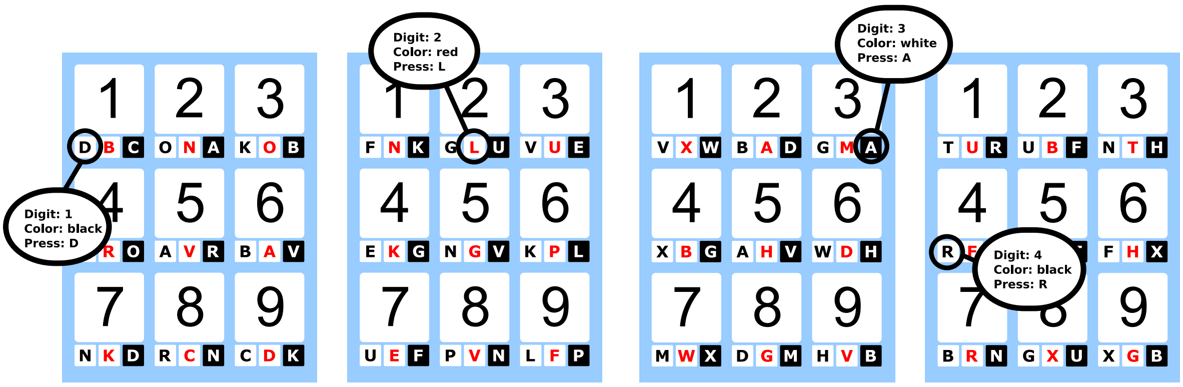


Figure 3.20: Exemplary PIN entry with ColorPIN. To input the PIN 1(black) 2(red) 3(white) 4(black) the user inputs the letters "DLAR". After each key press, letter assignment changes randomly as can be seen in the different steps.

user presses a key. Each letter that is assigned to the keypad occurs in all the three colors. For instance, in figure 3.20 (left), the letter "D" can be found at the bottom of digit "1" in black, at the bottom of digit "7" in white and at the bottom of digit "9" in red. These design choices were made due to security reasons which will be explained later.

To enter a digit of the PIN, the user has to input the letter that is displayed at the digit's bottom in the respective color. Input is done on a standard keyboard with one key per letter and not on the screen directly. Figure 3.20 exemplarily outlines a possible interaction to input the previously mentioned colored PIN. To enter the digit 1 (black), the user inputs the letter "D". After each step the letters are randomly reassigned. To input the second digit 2 (red), the user inputs "L". Finally, the user inputs "A" for 3 (white) and "R" for 4 (black). Therefore, the actual indirect input – which is the only thing that an attacker can observe – consists of the character sequence "DLAR" with which the user is successfully authenticated to the ATM. This way, a one-to-one relationship between the required button presses and the length of the PIN is preserved. Due to the random letter assignment, the character sequence is very likely to be completely different the next time a user authenticates with the system.

Prototype

As one of the advantages of software based authentication mechanisms, the implementation of the prototype only required writing a simple software. Since ColorPIN is a very graphical approach that does not require any complex network communication or the like, the software was written in Adobe Flash. The prototype allowed the users to define their own colored PINs as well as use randomized predefined PINs that were stored in an XML file.

Based on a preliminary paper prototype and several brainstorming sessions, the final user interface looked as depicted in figure 3.20. It had a light blue neutral background which was used instead of a white background since white was needed as a background for the colored letters and the digits which had to be clearly identifiable. The letters are located directly underneath the digits. In an early stage, a digits-only version was considered which would have made the software compatible to any terminal that has a keypad. This version was however rejected due to high confusion that it created by replacing digits with digits. As for standard PIN-entry, the only feedback that was given were asterisks in an input field. For input, the prototype used a standard commercial keyboard. Using a standard ATM setup, after three tries, the entry was stopped and the authentication session was considered as failed. In preparation for the user study, all interaction with the system was logged and stored together with an ID that had to be input before each trial.

Theoretical Security Analysis

Since for ColorPIN, not only the digits themselves but also their colors have to be remembered, its theoretical password space is 531,441 (27x27x27x27) for a four-digit PIN. This is based on the requirement that a user does not only have to input the right PIN but also use the correctly colored letters for it. For instance, in the example in figure 3.20, entering "BEGF" would represent the same PIN but would be rejected by the system since the letters have the wrong colors. This makes it much more resistant than standard PIN to a number of simple attacks. For instance, educated guessing attacks are hardly successful. Even if a user chooses to take her birth date as the four-digit PIN and an attacker knows about this, there is still the secret information about the color, which is way harder to guess.

Besides this basic security enhancement, the concepts discussed in the previous section make the system resistant to more elaborated (and more dangerous) attacks as discussed in the threat model:

Indirect Input through Letters Indirect input consisting of letters is an effective counter measure against shoulder surfing as well as camera recordings of the keyboard or the installation of fake keyboard hardware. Even if an attacker can see or record the whole input, the real colored PIN remains hidden. Additionally, it makes the system resistant to attacks based on Trojans and other spy software. Therefore, it would theoretically be appropriate for securing online banking and the like as well.

Using indirect input, the security of the system is (almost) completely independent of the user. While the user can still become a security problem by telling the PIN to someone or writing it down, the PIN cannot be disclosed by "insecure behavior" during the input at the ATM. This is a lesson learned from the VibraPass user study.

Reoccurrence of Letters As mentioned before, each letter used in the interface occurs in each color and therefore as a representation for three different digits. That is, even if an attacker can record the whole input as well as the screen (in its four appearances) the actual colored PIN is still hidden in a set of 81 (3x3x3x3) possible PINs.

Reassignment of Letters after Key Press After each input, the letters at the bottom of each digit are randomly generated and reassigned. Without this measure, an attacker could simply start the authentication with a terminal over and over again without actually trying to authenticate until the exact same screen layout as during the attack is depicted. Randomly reassigning the letters for each digit of the PIN renders this attack useless.

Main Weakness The main weakness of the system is an intersection attack. If the entry can be completely recorded several times in a row, the PIN can be stolen based on intersections between the observations. In most cases, two perfect observations lead to breaking the PIN. With each observation, the attacker collects 81 possible combinations out of the 531,441 possible PINs. In the two collected data sets, at least one PIN – the correct one – intersects. The possibility of having additional intersections in the rest of the data set is of interest for the security of the system. The more intersections, the higher the security of the system. As shown in equation (3.1), the chance of having one intersection in addition to the real PIN is already close to 0% and thus the chance of breaking the system in two attacks is nearly 100%. It is an interesting observation that the huge password space of ColorPIN is the main reason for this weakness of the system. The smaller the space, the lower the chance for intersection attacks. In a real world setting however, we consider it very unlikely to perform two perfect attacks on the same user. Additionally, the huge password space has more advantages than disadvantages and thus remains desirable. Furthermore, the fact remains that it is almost impossible to steal the PIN in a one time attack.

$$\frac{1}{\binom{531440}{80}} \times \binom{80}{1} \times \binom{531440-80}{80-1} \approx 0 \tag{3.1}$$

Usability and Security Evaluation

The evaluation of the system was conducted with the Flash prototype. The study set up consisted of a standard desktop PC with a standard commercial keyboard attached to it. That means, a keyboard with one key per letter. Additionally, a keypad as known from ATMs was connected to the PC to be used for standard PIN-entry. The whole interaction and every single key press was logged for later analysis. A camera was installed, filming the keyboard and the keypad as well as the screen. The filmed material was used to analyze user behavior as well as to simulate an attack on the system.

User Study Design ColorPIN was evaluated using a *repeated measures within participants factorial design*. The independent variables were *password type* (random ColorPIN or PIN and user generated ColorPIN or PIN) and *authentication mechanism* (standard PIN and ColorPIN). Standard PIN-entry represented the control condition to compare the performance to. The task was to authenticate to the terminal using every combination of the independent variables (*authentication system* x *password type* = 4 authentication sessions). The order of the independent variables was counterbalanced to minimize learning effects.

Procedure and Participants The lab study started with an introduction to the participants. The whole procedure, their rights as well as the prototype were explained in detail to each of them. Each participant got a random identification number which at the same time was used to assign the counterbalanced order of tasks to them. The study then started with a questionnaire collecting basic demographic data as well as information about the participant's PIN usage behavior.

Subsequent to this, the practical part followed. Before each authentication mechanism, the participants were asked to define their own ColorPIN and PIN. At the same time, one randomly generated PIN and one ColorPIN were assigned to them. Before using the system, they were explained in detail to the participants followed by a short training phase with one successful authentication. For each authentication session, there was a maximum of three tries to authenticate correctly. Switching to the next authentication session took place after successful authentication or if the attempts failed three times (which did not occur in the study). Following each authentication mechanism, the participants were asked to fill out a questionnaire containing questions about the respective system. In the end, a final questionnaire was handed out to them collecting comparison data about the systems. Ratings were given using Likert scales from 1 (disagree) to 5 (highly agree). Every key press, correction, error etc. were logged by the prototype.

24 volunteers participated in the study. The average age was 28 years, the youngest participant was 15 and the oldest 57 years old. 18 of them were male, six were female. Choosing 24 participants allowed for perfectly counterbalancing the independent variables to minimize learning effects. Having 24 participants, the results of the study are based on 96 authentication sessions.

Hypotheses The following main hypotheses were stated for the user study. ColorPIN is:

(H1) more secure to observation attacks than standard PIN-entry.

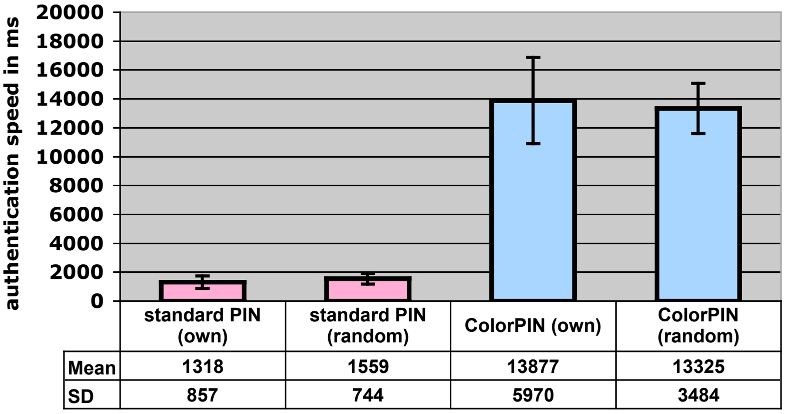
(H2) more error-prone than standard PIN-entry.

(H3) slower than standard PIN-entry.

Results

Authentication Speed Authentication speed was measured for each authentication session from the first key press (entering the first digit of the PIN respectively the first letter) to releasing the last key. This decision has been made to compare the actual interaction times since pressing an ok button takes the same time no matter which method is used and would be an unfair advantage for ColorPIN in comparison to standard PIN which is supposedly faster.

Only successful authentication attempts were taken into account for the analysis. The average authentication speed for the authentication mechanisms is shown in figure 3.21. Standard PIN-entry using a user generated PIN was the fastest (M=1.32s, SD=0.86s), followed by random standard PIN (M=1.56s, SD=0.37s). ColorPIN with a user generated PIN was the



	standard PIN (own)	standard PIN (random)	ColorPIN (own)	ColorPIN (random)
Mean	1318	1559	13877	13325
SD	857	744	5970	3484

Figure 3.21: Average authentication speed of the ColorPIN user study. The results show that with around 13 - 14 seconds, ColorPIN is slower than standard PIN-entry.

slowest method (M=13.88s, SD=5.97s) and slightly slower than ColorPIN with a random PIN (M=13.33s, SD=1.74s). A 2 x 2 (*authentication mechanism* x *password type*) within participants analysis of variance showed a highly significant main effect for authentication mechanism ($F_{2,46}$=64.50, p<.001). No significant interaction effects and no significant main effect for password type were found. Considering these results, hypothesis (H3) can be accepted.

Error Rate Following the standard approach as used for the other authentication mechanisms presented in this thesis, basic and critical errors were distinguished within the error analysis. Thus, during the study, we measured whether a participant could correctly authenticate to the system within three tries and how many corrections (deleting the input) were required to do that. For standard PIN-entry, each participant could successfully authenticate to the system at the first attempt, no matter if a random PIN or a user generated PIN was used. Only two users (random PIN) respectively one user (own PIN) applied corrections to the input. Regarding ColorPIN, two users for each password type needed two or three attempts to authenticate. However, no authentication session resulted in a critical error. Six users (four random PIN and two user defined PIN) needed to use at least one correction to authenticate. The data revealed no significant differences neither between the different authentication mechanisms nor the different password types. Therefore, (H2) cannot be accepted.

User Satisfaction Even though the error rate of ColorPIN was as low as for standard PIN-entry, it was almost ten times slower. In the questionnaire, participants were asked to rate the speed and many other aspects of the two systems on Likert scales from 1 (disagree) to 5 (highly agree). As to whether the systems were fast, standard PIN (M=4.7) was rated faster than ColorPIN (M=3.4). However, the rating for ColorPIN still has a positive tendency, which is surprising. A similar result can be found when comparing the ratings for whether the systems were easy to use. Again, standard PIN (M=4.8) was considered easier than ColorPIN (M=3.2) but the tendency remained positive. Surprisingly, asked whether they would like to use the systems in

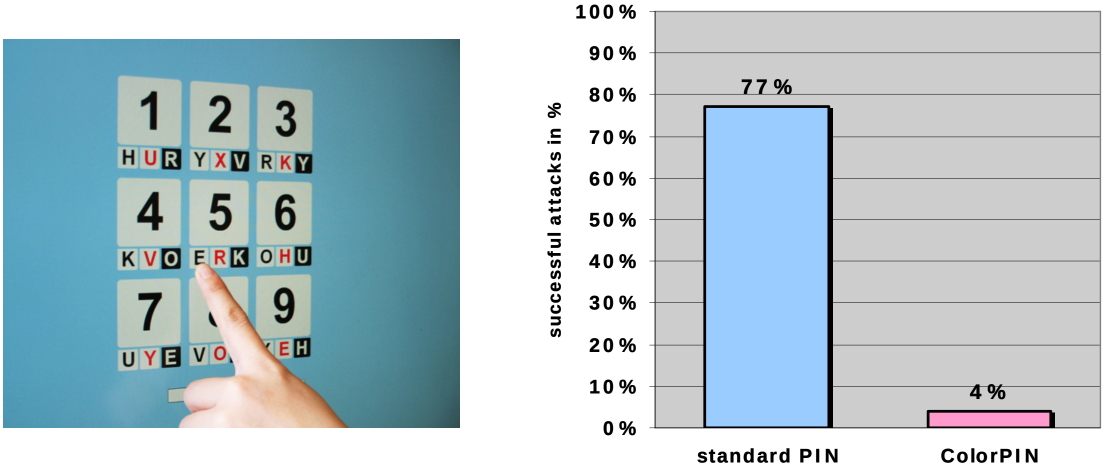


Figure 3.22: Left: Two participants pointed at the screen during input which enabled the attacker to steal their ColorPINs. Right: Number of successful attacks on both systems based on the video material.

real world scenarios, standard PIN (M=3.5) and ColorPIN (M=3.6) were rated nearly equally. The positive qualitative results for ColorPIN – contradicting the quantitative measures – are very likely influenced by effects of the laboratory setting as well as novelty of the interaction but still encouraging.

The questionnaire revealed some design flaws that might have negatively influenced performance of ColorPIN. For instance, some users mentioned problems with the choice of colors. One user stated that “the colors, especially of the different backgrounds are confusing‘. A unified background color might help finding the letters faster.”. That user was refering to the fact that the background of white characters was black and white for black characters. This seemed to confuse several of the users and negatively influenced the speed of the visual search task. In a future version, this problem should be fixed.

Security Besides the theoretical security analysis, the camera material collected during the study was used to perform a practical security evaluation and to find out whether ColorPIN was more resistant to a camera attack than standard PIN-entry. The attacker was familiar with the ColorPIN system and had the full video material available as recorded during the study. The sound layer was removed to avoid disclosure of authentication tokens due to conversations during the study. As opposed to VibraPass, sound effects on security were ruled out. The attacker had three tries to authenticate correctly. If the correct authentication token was entered within these three attempts, an attack was counted as successful.

Out of the 48 authentication sessions using standard PIN-entry, 37 (77%) could be successfully identified (see figure 3.22, right). The remaining ones were mostly cases in which the participants were hiding the PIN entry with the non-active hand. As opposed to the VibraPass study, the approach was not a worst-case scenario with several cameras but an attack that is more common on

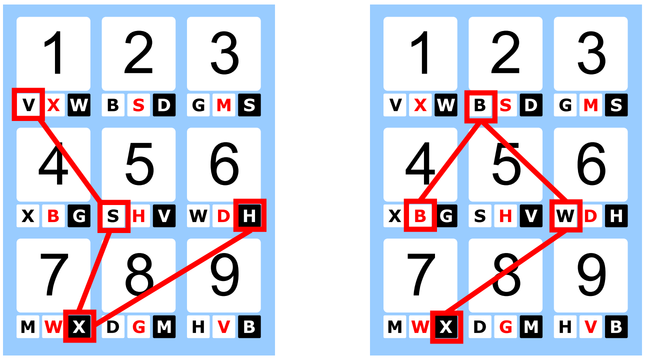


Figure 3.23: Some users applied a simple memory and performance trick when using ColorPIN. Instead of remembering the color-coded PIN, they remembered the locations where the characters appear, together with a respective path.

public terminals, using one camera. However, as a lesson learned it can be highlighted that evaluating the worst-case always seems to be the better choice. Out of the 48 ColorPIN authentication sessions, only two of the different authentication tokens could be identified (for two different users). In both cases, the users were pointing (at the screen) at the characters they wanted to input to assure they were choosing the right letter as shown in figure 3.22, left. This kind of behavior cannot be referred to as the system requiring the user to behave “clever” as we observed it for the VibraPass systems but should be considered as insecure behavior that can break the security of a system [2]. With respect to this data, hypothesis (H1) can be accepted. Additionally, this assumption is supported by the questionnaire in which the participants considered ColorPIN (M=4.6) more secure than standard PIN (M=2.6).

Discussion

The evaluation showed that using its special way of indirect input, ColorPIN is notably more secure than standard PIN-entry. At the same time, the indirect input creates extra cognitive load which makes the system significantly slower. Even though the results of the study might have been negatively influenced by design issues (choice of colors etc.) and the short training phase, they are already promising in comparison with related software based authentication systems like [59, 106, 124, 135]. Furthermore, we can informally state that after repeated use of the system it becomes remarkably faster. In an informal study, trained participants achieved average times of about 3.5 seconds from the fifth authentication session onwards.

A very interesting, but also informal observation was on strategies that improved performance. Some users reported to having remembered the positions of the characters instead of the colored PINs. This way, they exploited the fact that even though the characters are randomized, the positions in which they appear remain the same. Therefore, they constructed invisible paths on the visualization as shown in figure 3.23. These shapes are similar to findings from the EyePIN study (see figure 3.5) and can both help to improve performance as well as memorability.

The additional information “color” and the lack of exploiting the users’ muscle memory for input (since the movements required for input on the keyboard change every time the user authenticates), can lead to decreased performance with respect to memorability. This is supported by the opinion of a user who stated that “I suppose it is harder for me to remember a PIN including colors than just a PIN”. As mentioned before, we could observe new interaction strategies that might solve this memorability problem.

The most interesting part of ColorPIN is that a simple trick (using colored PINs) made the system resistant to camera attacks. No comparably secure software based authentication system in related work achieves such good performance results. Only systems that are secure against shoulder surfing only can compete with ColorPIN speed-wise. Based on the lessons learned from the VibraPass evaluation, ColorPIN was the only authentication mechanism in this thesis that was specifically designed and implemented with the criteria in mind that security should not be based on “clever” behavior of the user, even though others fulfill it as well. This goal was successfully achieved.

3.5 Lessons Learned

The authentication mechanisms presented in this chapter were designed with the goal to solve problems of their specific categories and thus to present candidates that might (with modifications) replace standard PIN or password entry in specific scenarios. But more importantly, a lesson learned was that candidates that look good can disqualify themselves by applying stricter criteria than standard usability evaluation. One very good example are the findings on “bad lies” that showed that special issues that are way out of the scope of normal usability criteria can negatively affect the security of an authentication mechanism.

Keeping this in mind, evaluation of authentication mechanisms (especially for public spaces) has to be reconsidered and adapted. One of the first “new” criteria that was learned within the scope of this thesis was that security should not require an active user. That is, the important property “security” should not rely on how clever the user handles or uses the presented system but should be implicitly built-into the system so it performs equally secure for all users.

Another important lesson was found during the evaluation of user owned device based approaches. As known from related work, time is mostly reported very imprecise, sometimes including confirmation procedure, sometimes omitting such phases. Oftentimes, it is not clear at all what has been measured. Besides this, MobilePIN taught us that it is of high importance to “honestly” measure and report input speed. For instance, not reporting connection establishment time can make a system look really fast. Considering that in most scenarios a connected mobile device is not required (not part of the scenario itself), this means that in these cases the time for establishing the connection has to be counted as part of the authentication process and thus also reported as part of it. That is, authors either have to explicitly state the scenario in which the system should work or admit that it is rather slow.

Having a look at the practical security evaluations of the systems in comparison to the theoretical approaches (which is often the only information reported in work on authentication mechanisms), one can see that theoretical high security does often not hold against real conditions – and most importantly against users. Using perfect surveillance (worst-case scenarios) often reveals weaknesses that cannot be found in a theoretical analysis. Therefore, the lesson learned here is not only to use the hardest attack possible but also never to rely on theoretical results. As seen in the work on ColorPIN, a very large password space can eventually become a security issue even though usually considered a property of high security.

Overall, this means that there is a gap in evaluating authentication mechanisms that has to be closed. The question is what the criteria are that define usable and secure system really are. Learning from "bad lies", one conclusion is that behavioral factors have to be considered more. Moreover, standard criteria like security or authentication speed have to be reconsidered. To gather more knowledge in this open area, in the following chapters, work on long-term evaluations of authentication mechanisms (chapter 4) and field-studies (chapter 5) will be presented that helped to identify important criteria.

Chapter 4

Evaluating PIN and its Influence on Standard Usability Factors

Begin at the beginning and go on till you come to the end: then stop.

– The King of Hearts (Alice's Adventures in Wonderland) –

This chapter will teach us two things: Firstly, it will show us how going back to the beginning can prove very helpful. Second, the work described in this chapter was the main reason why we redefined speed measurements for authentication mechanisms. It will tell us how and where to begin measuring time, how to come to an end and most importantly, where to stop.

The work on advancing authentication mechanisms gave diverse indications that standard usability factors, especially time, might have to be reconsidered and finer-grained to allow for more insightful evaluations. To understand how this might look like, we decided to take a step back and take a detailed look at the basic form of authentication mechanisms. A long-term evaluation of standard PIN-entry in different configurations seemed to be the appropriate tool for this undertaking since they often reveal different results as compared to one-time lab studies and provide more insights, as, for instance, shown by Chiasson et al. [16]. Additionally, using standard PIN-entry, which users are familiar with, the study allowed for evaluating basic interactions rather than novelty factors of any other proposed system. The contributions of this chapter are based on the results of this evaluation[1]:

a) It showed how evaluating basic factor of a system can reveal interesting findings and advance the field.

b) The study highlighted the importance of consistency, especially for a short and highly focused task such as authentication.

c) Finally, the results were used to create a time measurement concept covering several phases that we believe should be applied when evaluating the speed of authentication mechanisms. The study showed that applying this measurement can lead to important findings that would have remained hidden otherwise.

4.1 Not just a Keypad

In the early fifties, rotary dials were the standard on telephones. Since that time, significant research effort was spent on improving the usability of the dialing experience. This effort finally led to the introduction of the standard telephone layout as it is found nowadays on most mobile and land line phones worldwide.

When they planed to replace rotary dials with pushbuttons, Bell Telephone Laboratories invested a lot of research on finding the "right" layout. Lutz and Chapanis [86] found out in a big study that most people consider the keypad as we know it today as the most convenient and natural layout. That is, digits are ordered from 1 to 9 in a 3x3 matrix, starting in the upper left corner. The zero is located in the middle of the fourth row. Follow-up studies could never attest a significant advantage of this so-called telephone layout compared to other layouts like the calculator (1 in the lower left corner) but always found a subjective preference that finally led to its introduction for modern phones [22, 38, 56, 91]. In these studies, many different layouts were tested. The only one that is slower than the telephone layout, with significant results, is the random layout, as a recently shown in a study by Ryu et al. [109]. Memorability studies by Conrad et al. [21] gave further evidence that unnatural layouts like the random layout might have drawbacks. In their study, they could show that such an unnatural layout led to significantly more errors that could be tracked back to memorability issues.

This list shows the amount of work that was put into the introduction of they keypad as it is used for so many tasks nowadays, including PIN-entry. Despite all this work, it is very surprising that there is still no official standard for the design of keypads for public terminals. Even though recommendations exist[2], service providers are still free to design their public terminals however

[1] This chapter is partially based on [130]. The practical part of this work was conducted by Emanuel von Zezschwitz together with the author of this thesis.

[2] e.g. by the European Committee for Banking Standards

they want. Therefore, in practice, telephone layouts as well as calculator layouts and even linear layouts (all numbers ordered from 1 to 0 in a horizontal row) can be found. Patents like [128] show that random layouts are considered by industry as well. Furthermore, randomization is often the tool of choice when it comes to securing authentication mechanism.

4.2 Long-Term Study

The evaluation described in this chapter is based on all this work and thus compares all the previously mentioned keyboard layouts to each other. More than that, it goes one important step further. Instead of comparing their performance within lab experiments based on randomly generated one-time PINs, we decided to assign a fixed PIN to each user and evaluate it in a long-term study over several weeks. This way, a more realistic setting was chosen which allowed for better insights on the "true" performance costs of the different layouts. This set up also enabled us to test on memorability effects over time.

That is, the here discussed study presents an evaluation of the influence of different keypad layouts on authentication performance. Standard PIN-entry seemed to be the appropriate choice for this approach. Furthermore, a much more in-depth authentication speed measurement approach was used. This was done as a consequence of the lessons learned from the evaluation of the authentication mechanisms introduced in chapter 3. This way, additional consistency problems could be identified and the importance of such a measurement was highlighted. It also resulted in a very detailed approach for measuring speed of authentication mechanisms and impressively showed its effectiveness in revealing important and interesting facts.

4.2.1 User Study Design

The user study was designed as a repeated measures longitudinal experiment with one independent variable, *keypad layout*, with four levels as shown in figure 4.1:

Telephone layout This layout consists of a 3x3+1 matrix of digits. The digits are sorted ascending, starting in the upper left corner and ending in the lower right. The 0 is located underneath the 8. This represents the most deployed keypad setup at American and European terminals.

Calculator layout The calculator layout is very similar to the telephone layout with the exception that the first and last rows are swapped. This layout can found on QWERTY-keyboards, calculators and at many public terminals in Asia.

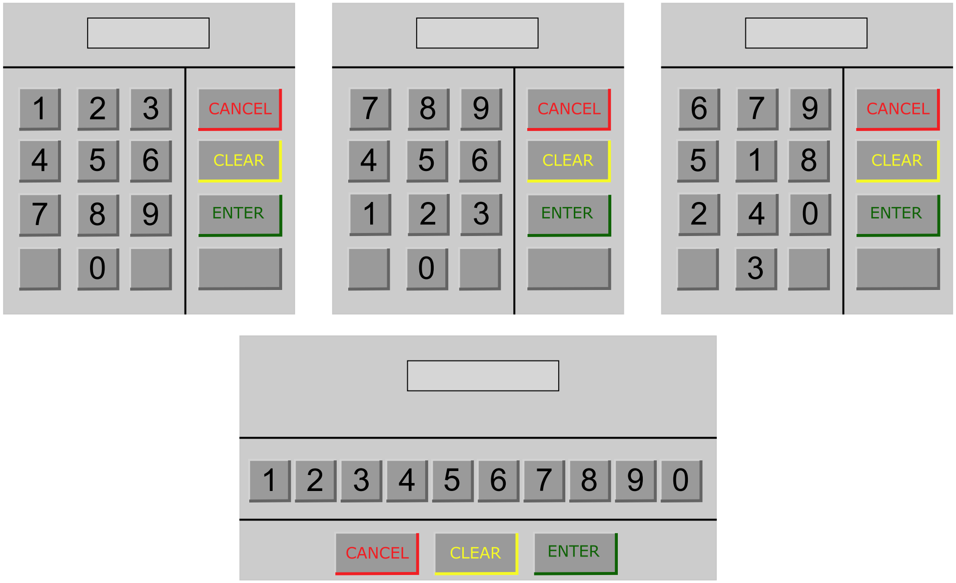


Figure 4.1: Keypad layouts used in the long-term study. Top, left: Telephone layout. Top, center: Calculator layout. Top, right: Example of a random layout. Bottom: Linear layout.

Linear layout The linear layout looks like on a standard commercial QWERTY-keyboard. The digits are arranged in one row, starting with 1 and ending with 0. Therefore, this layout represents a standard condition that users are often confronted with, not only when using personal computers but also when interacting with public terminals, again, mainly in Asia.

Random layout Finally, the matrix of the random layout is again a 3x3+1 layout. The digits are randomly assigned to their position for each authentication attempt. The reason why it makes sense to evaluate random keypads is that they are considered more secure against shoulder surfing and skimming attacks. Furthermore, they represent an approach that is often used in related work to obfuscate and secure the authentication process (see for instance [135]). The results of this study will show why, to some extent, randomization can be considered harmful for the usability of an authentication mechanism.

The dependent variables measured were authentication speed, error rate and memorability. Authentication speed has been measured with a focus on having very precise and structured measurements and to include all possible times as opposite to the standard approach taken for the usability evaluation of authentication mechanisms. This will be explained in more detail in the next chapter.

Each participant was exposed to every layout in counterbalanced order to minimize learning effects and other unwanted influences of the layouts. Such a design is very robust against variations across different groups of people.

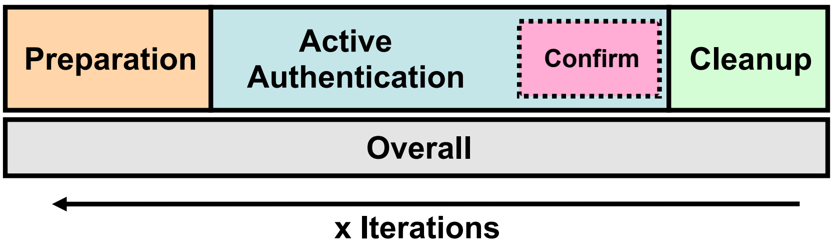


Figure 4.2: The different phases involved in the extended authentication speed measurement as applied in the user study.

4.2.2 Phase-based Time Measurement

To understand the time measurement as it was performed within this long-term user study, we firstly have to introduce the approach in detail as shown in figure 4.2. We divided the authentication process in four different phases that will be explained later. From related work, we know that these phases are sometimes implicitly recorded but for unknown reasons never make it to the analysis part (see for instance Chiasson et al. [16]). The results of this study will highlight that it is necessary to consider these phases and their respective times.

Preparation phase contains every interaction and time that is spent from the moment that the users could theoretically start the authentication process to the moment that they actually start it. Of all the times in this concept, this one is the hardest to measure since it is not trivial to say when that moment actually takes place. In this long-term study a “trick” based on a countdown has been used that will be introduced later.

Active authentication is the time that it takes to perform the active tasks (pressing buttons and the like) required to successfully authenticate. Since in many cases it can be considered “unfair” to include confirmation times (like pressing the “enter” button) into the comparison with other systems, this time should be measured separately as a sub-task of active authentication. Unfair refers to the fact that pressing an OK button takes the same amount of time, no matter how efficient the system works. That is, a slower system might unfairly profit from this phase if it is reported as part of the active authentication phase.

Cleanup includes any task that has to be performed to be able to start with the actual task. For instance, it could include removing specific authentication hardware or physical tokens. Depending on the authentication mechanism, this phase might be missing.

Finally, *overall time* consists of the sum of all phases. Based on the assumption that the authentication token can be input wrongly, several iterations (up to three in case of an ATM) of the phases can be required. Depending on the authentication mechanism, preparation and cleanup might not play a role during iterations.

4.2.3 Procedure

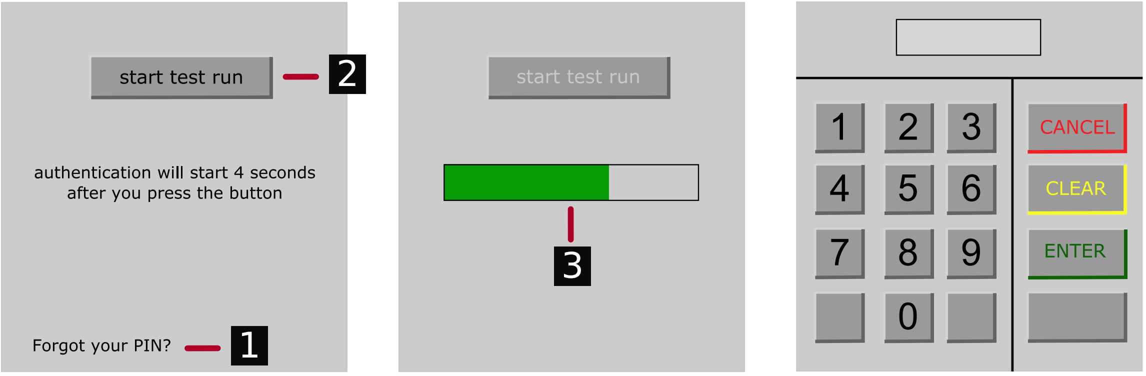
60 volunteers finished the longitudinal experiment in a little less than two months. On the first day of the experiment, each participant was asked to answer a questionnaire containing 29 questions. Besides gathering demographic information, the answers were mainly used to identify "experts" within the group of participants. An expert user was defined by withdrawing money at an ATM at least two times per week. The other users took money around three times per month.

The task was to authenticate to an online version of an ATM two times per week. In most cases, the telephone layout was used but in different intervals the other layouts were injected into the trial. Each special layout was injected exactly two times for each participant. Resulting in six authentication attempts with layouts other then the telephone layout. A 6x6 Latin square design was used to inject the special layouts to minimize effects of the different layouts. Besides the six authentication attempts with the special layouts, another eleven attempts were performed with the telephone layout. That is, overall, each participant performed 17 trials throughout the study. The first day using the telephone layout was considered as the introduction to the system and training phase and was therefore not included in the analysis.

The "expert" group performed an initial week in which they authenticated with their PIN every day on a telephone layout. That is, those participants performed another seven authentication attempts before starting the actual study. Using an online study does have several limitations like lack of control. However, using such a big group over a period of two months, the study could not have been handled otherwise.

In the implementation of the prototype system, we placed particular importance on correct time measurement. Thus, in both groups, all different phases were recorded as proposed in figure 4.2. The keypad was simulated using software written in Adobe Flash CS4 which was embedded in a website. Using an online study, the participants could complete the tests independently of the place. The keys of the keypad could be controlled with any pointing device or touch screen (the input device did not significantly influence the results). Interaction with the keyboard, however, was not possible since the labeling of the keyboard would have influenced the test performance too much. The simulated keypad had an ATM like design with the number keys positioned in the given layout and the three function keys cancel (red), clear (yellow) and enter (green). The different layouts, as displayed in the prototype, are depicted in figure 4.1. The system logged all user interactions, interaction times and error rates.

On each study day, the users got an e-mail at 00:10 am informing them to perform the study on that respective day. This e-mail only contained the participant's ID and a link to the study website. To prevent unintentional training effects, the participants had to perform the authentication on exactly that day. If a participants did not perform the study till 6:00 pm, a second reminder e-mail was sent. If a user did not perform the authentication within one day, an alternative day was scheduled to perform the trial. However, to avoid negative effects on the results, a maximum of two alternative time slots per participant was allowed. Missing the trial more than two times resulted in the participant being excluded from the study. During the expert training phase, no alternative time slots were allowed.



authentication will start 4 seconds after you press the button

Figure 4.3: User interface of the long-term PIN study software. Left: 1. In case the participants forgot their PIN, they could look it up. 2. Button to start the authentication process. Center: 3. The study countdown. After pressing the button, a four seconds countdown appeared. This was used to define an exact starting point of the interaction to count preparation time. Right: An exemplary keypad layout.

To authenticate to the system, users logged in to the test environment using the ID provided in the reminder e-mail. If they had forgotten their PIN, they had the possibility to review it within the system. This was designed to provide a theoretical possibility to log memorability problems. However, from the questionnaires we know that many participants simply wrote down their PINs and used those notes instead of the online tool to look up their PINs as shown in figure 4.3, left. When the users pressed “start test run” (see figure 4.3, left), a four second countdown appeared as shown in figure 4.3, center. After the countdown, the keypad appeared and the authentication could begin. This countdown allowed us to record the time for preparation. Preparation measurement started at the end of the countdown and ended with pressing the first button. A short countdown was chosen to avoid participants using it for preparation like thinking about the PIN. According to the configuration of real ATMs, a maximum of three incorrect entries was allowed. Entries could be corrected as often as desired by “cancel” and “clear”. At the end, an entry had to be confirmed by pressing “enter”. Finally, after each trial, the participants had the possibility to leave comments using a small text box. They were told to use it to report problems they encountered during authentication and to share their thoughts with us. After performing the last authentication attempt, the participants were asked to fill out a final questionnaire which concluded the study.

4.2.4 Participants

Participants were recruited over social networks, different mailing lists and university related bulletin boards. Finding volunteers for a two months experiment is a rather difficult task. To motivate participants and to keep their motivation up, all participants that finished the experiment could win some prices. This seemed to work quite well. Of the 66 users that started the study, only six dropped out, which is less then 10%. The average age of those 60 participants was 24, ranging from 14 to 33. 37 were male, 23 were female. As mentioned before, some participants were defined as “experts” or “trained” users. In the end, 29 trained users and 31 untrained users finished the study.

An interesting finding was that many participants did not seem to consciously perceive different keypad layouts in everyday use. For instance, 11.7% did not know that the standard ATM keypad in Germany is equal to the standard telephone layout. Another 31.7% thought that the standard keypad layout at ATMs was equal to the numpad layout on a standard commercial keyboard. 63.3% of the participants mentioned that they would use the linear layout on their keyboard whenever they have to enter PINs using their computer. This high number can partially be explained by the wide spread of laptop computers that seldomly provide adequate numpads. This also indicates that study participants were very familiar and accustomed to the linear layout.

4.2.5 Follow-up Study

The main study revealed interesting findings, many related to learning effects in the trained group. When comparing the data of the telephone layout to the random layout data, an interesting effect of our time measurement approach was observed. However, since the random layout data was only based on two authentication attempts per participant, a stronger data set was required. Therefore, a follow-up study was conducted. The conditions were the same as in the training phase of the main study with the difference that the participants used the random layout instead of the telephone layout. Each participant authenticated once per day over a period of seven days.

We recruited eleven participants with an average age of 27 years. The youngest participant was 22, the oldest 42 years of age. Eight participants were male, three were female. None of the volunteers participated in the original study. The demographic data of the follow-up study did not significantly differ from the main study. The lower number of participants in the follow-up study could lead to less accurate results and a stronger influence of outliers. Eleven participants, however, are enough to consolidate the results of the main study. The accuracy of the results is strengthened by the high number of repeated measures.

4.2.6 Hypotheses

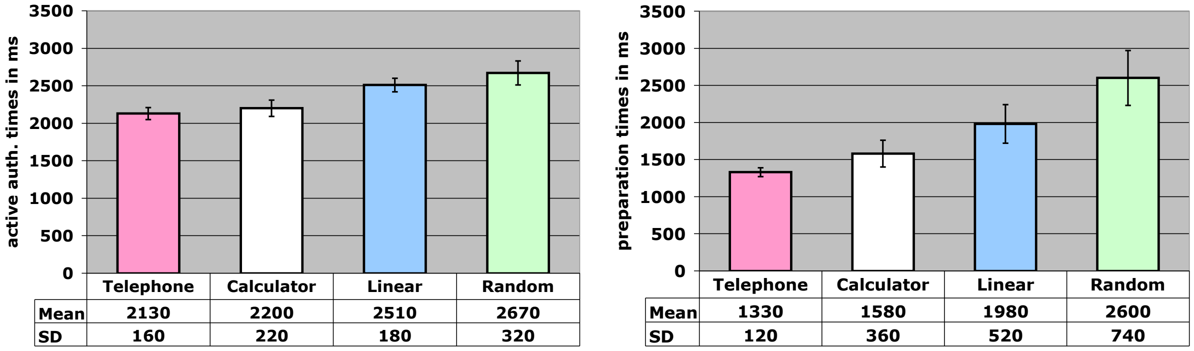
Based on experiences with evaluating PIN-entry, the following main hypotheses were stated:

(H1) Using alternative keypad layouts[3] has a negative influence on memorability.

(H2) Using alternative keypad layouts has a negative influence on error rates.

(H3) Using alternative keypad layouts has a negative influence on authentication speed.

[3] An alternative layout in this study refers to a layout that is different from the main layout used by the participants. That means either calculator, linear or random layouts.



	Telephone	Calculator	Linear	Random
Mean	2130	2200	2510	2670
SD	160	220	180	320

	Telephone	Calculator	Linear	Random
Mean	1330	1580	1980	2600
SD	120	360	520	740

Figure 4.4: Left: The average times of the different layouts required for the active authentication phase. Right: Average times for the preparation phase. The graphs show that preparation time plays an important role when measuring the performance of an authentication system.

And finally, based on the follow-up study:

(H4) Using a consistent layout allows for a significant learning process, while using the random layout does not.

4.3 Results

4.3.1 Authentication Speed

Figure 4.4 depicts the average preparation and active authentication speed for the different layouts. These numbers reported are based on the data collected during the main part of the long-term study. The first day of the main phase was removed since it was counted as the introduction to the system. Therefore, the results are based on 600 authentication sessions with the telephone layout as well as 360 authentication sessions with the alternative layouts (120 per layout). Moreover, only successful authentication attempts were considered in the analysis.

As shown in figure 4.4, left, the telephone layout requires the shortest time for the active authentication phase (M=2.13s, SD=0.16s). Using the calculator layout takes 3.3% and the linear layout already requires 17.8% more time. As expected, the biggest overhead is created by the random layout (25.4%). A one-way ANOVA revealed a significant main effect for *keypad layout* ($F_{1.95,114.96}$=9.13, p<.01, Greenhouse-Geisser corrected values). Tests of inner subject contrasts showed a significant advantage of the telephone layout in comparison with the linear ($F_{1,59}$=31.63, p<.01) and the random layout ($F_{1,59}$=16.42, p<.01). The differences of the calculator to the linear ($F_{1,59}$=8.37, p<.05) and random layout ($F_{1,59}$=8.7, p<.05) were significant as well. No significant differences between the telephone and the calculator layout as well as between the linear and the random layout were found (all p>.05).

Having a look at the results of the preparation phase (figure 4.4, right), reveals a similar trend. However, the negative effects of unfamiliar layouts are even graver in this phase. Again, the telephone layout requires the shortest time on average (M=1.33s, SD=0.12s) followed by calculator (M=1.58s, SD=0.36s), linear (M=1.98s, SD=0.52s) and random layout (M=2.6s, SD=0.74s). A one-way ANOVA revealed a significant main effect for *keypad layout* on preparation time ($F_{2.15,126.66}$=5.63, p<.05, Greenhouse-Geisser corrected values). Tests of inner subject contrasts showed significant differences in the times required for the telephone layout compared to linear ($F_{1,59}$=4.84, p<.05) and random layout ($F_{1,59}$=12.26, p<.05). The calculator layout shows no significant difference to the telephone and linear layout (all p>.05). Nevertheless, it shows a significant advantage compared to the random layout ($F_{1,59}$=4.84, p<.05). Again, no significant differences between the linear and the random layout were identified (p>.05).

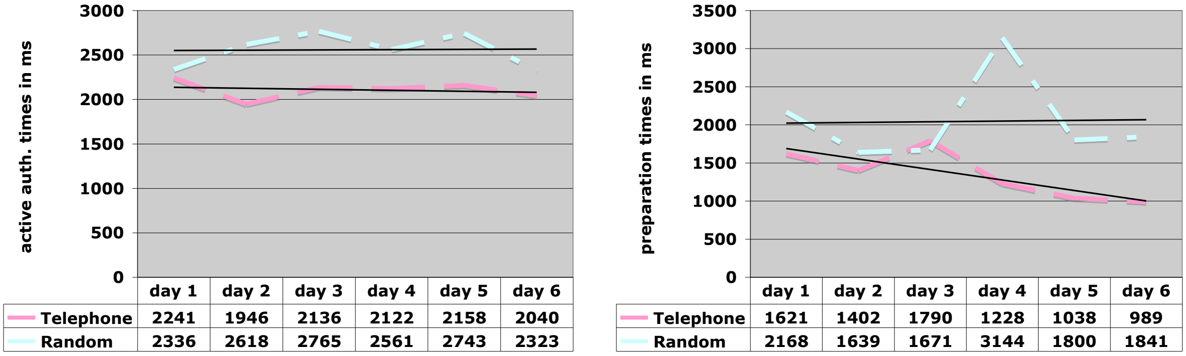
Interestingly, performance issues for the linear and the random layout were almost identical. The analysis showed that performance problems of the linear layout were mostly due to longer active authentication times. This can be explained with the bigger distances between the keys on the keypad. On the other hand, the random layout mainly suffers from prolonged preparation times due to lack of consistency.

While for all layouts the preparation phase is shorter than the active authentication phase, the random layout requires almost identical amounts of time for both. That is, the overhead added by the preparation phase is around 100% for this layout. Considering the time sensitive nature of authentication, this can be considered a serious drawback of random layouts.

The results showed that the more the layout differs from the one the participants are used to (telephone layout), the more time is required as well for active authentication as for preparation. This effect is even higher for the preparation phase. Based on these results, hypothesis (H3) can be accepted. The ratings of the layouts by the participants give additional confirmation for this hypothesis. The telephone layout was rated the fastest on a Likert scale from 1 (very slow) to five (very fast). 95% of the participants considered it very fast (79.7%) or at least fast (15.3%). The calculator layout is ranked second (28.8% very fast, 49.2% fast) followed by the linear layout (5.1% very fast, 18.6% fast). The random layout scored rather badly with 42.4% of participants rating it as very slow. Therefore, the subjective opinion of the users correlates with the quantitative results of the study. The quantitative differences become even clearer when taking an in-depth look at the differences between the telephone and the random layout.

More Telephone vs Random Layout

The follow-up study was performed to allow for a more precise comparison between the telephone and the random layout. Therefore, this analysis is based on the data of these two groups (trained telephone and trained random) over a period of seven days during which authentication was performed once per day. The first day counted as training and was therefore not included in the analysis. Authentication times are reported without the time required to confirm. This was done to avoid unfair disadvantages of the systems. Only correct authentication attempts were included in the analysis.



	day 1	day 2	day 3	day 4	day 5	day 6
Telephone	2241	1946	2136	2122	2158	2040
Random	2336	2618	2765	2561	2743	2323

	day 1	day 2	day 3	day 4	day 5	day 6
Telephone	1621	1402	1790	1228	1038	989
Random	2168	1639	1671	3144	1800	1841

Figure 4.5: Left: Active authentication times for the telephone and the random layout. Right: Preparation times show a significant learning effect for the telephone layout.

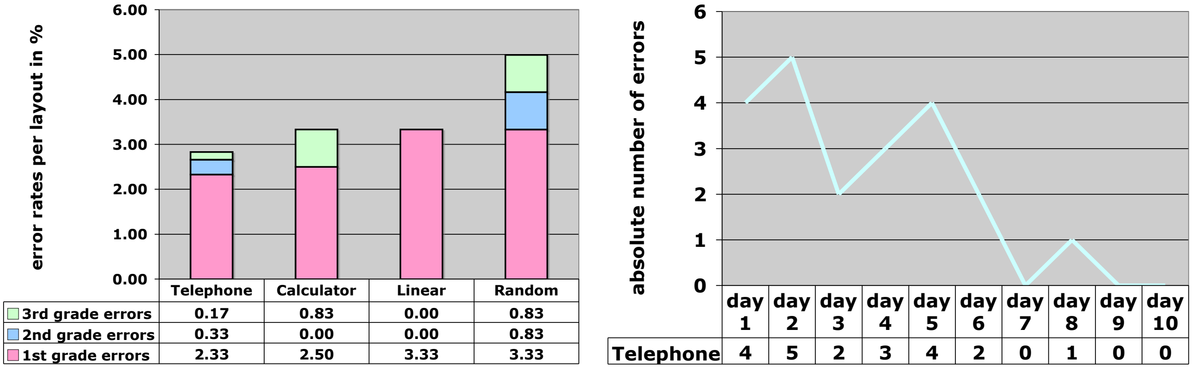
Figure 4.5 shows the comparison of the average times for the telephone and the random layout. The left graph shows the speed of active authentication. For the telephone layout, all times are between 1.94 and 2.24 seconds. We conducted an analysis of variance which indicated no significant main effects of days ($F_{3.6,100.6}$=0.85, p>.05, Greenhouse-Geisser corrected values). Authentication times of the random layout also show no effects of training ($F_{3.3,32.7}$=1.25, p>.05, Greenhouse-Geisser corrected values) and vary slightly from 2.32 to 2.77 seconds. Within the single test days, the time differences between the two layouts are not significant too (all p>.05). The black trend lines show how little effect exercise had on the times of both layouts. The linear trend lines are nearly parallel to the x-axis for both.

It should be noted again that in most studies, only active authentication time is measured and interpreted. Under the assumption that we had examined only this as well, we would have necessarily come to the conclusion that the additional work, which is associated with the use of a random layout, although measurable, is only marginal and stable. In addition, we would have to conclude that the lack of learning in the random layout is not a disadvantage, because the use of the telephone layout does not show any significant improvements as well.

However, taking a closer look at preparation times, which are shown in figure 4.5, right, it can be seen at first sight that the trend lines of both layouts are far apart. While there is no improvement for the random layout, the preparation times of the telephone layout constantly decrease with the exception of day 3. A test of the inner subject contrasts for the telephone layout showed that the times of the first ($F_{1,28}$=6.51, p<.05) and second day ($F_{1,28}$=9.31, p<.05) differ significantly from the time of the sixth test day[4]. Thus a significant learning effect was found using the telephone layout which was only revealed by evaluating the preparation phase.

This result is also reflected in the consideration of the overall time: preparation + authentication + confirm (no cleanup was measured since there was no additional interaction after confirm). By including the preparation time, we see that the constant use of a fixed layout in conjunction with repeated use can increase the authentication performance. A t-test showed that the times

[4] The "first day" refers to the second day of authentication since the real first day was considered as a learning phase.



	Telephone	Calculator	Linear	Random
🟩 3rd grade errors	0.17	0.83	0.00	0.83
🟦 2nd grade errors	0.33	0.00	0.00	0.83
🟪 1st grade errors	2.33	2.50	3.33	3.33

	day 1	day 2	day 3	day 4	day 5	day 6	day 7	day 8	day 9	day 10
Telephone	4	5	2	3	4	2	0	1	0	0

Figure 4.6: Left: Error rates for the different keypad layouts. Errors of 3rd grade indicate critical errors. Right: Error rates for the telephone layout over time. The graph contains critical as well as basic errors.

of the two layouts differ significantly beginning with the fifth day. The test results for the fifth, (t(38)=2.69, p<.05, r=.42) and sixth test day (t(38)=2.13, p<.05, r=.33) show significant differences of medium effect size. These results confirm hypothesis (H4).

The importance of measuring preparation time is additionally highlighted by the fact that in our experiment, it was for many participants longer than the active authentication time. This effect was especially high for the random layout. That is, actual overall time was up to 100% more in these cases. Since the preparation time is significantly different for both layouts, neglecting it leads to completely wrong conclusions.

4.3.2 Error Rate

The analysis of error rates for the different keypad layouts was based on authentication sessions of the main part of the long-term study. That is, the first week of the trained group was not included. The first authentication attempt was not part of the analysis either. With respect to the evaluation of authentication mechanisms as outlined in chapter 3, we distinguished between basic and critical errors. Basic errors indicate that an authentication attempt failed one or two times. A critical error occurred when the authentication failed completely (three wrong tries).

Surprisingly, the overall error rate for all conditions was very low. Out of the 960 authentication sessions (600 with the telephone layout and 360 with the alternative layouts), 929 did not result in any error (96.7%). 25 out of the 31 problematic authentication sessions resulted in errors of first grade. That means that they were successful in the second try. Another three failed for a second time and only three created critical errors. That means that only three out of 960 authentication sessions (0.3%) could not be completed successfully. Each layout besides the linear layout triggered exactly one critical error. Due to this low error rate, no significant effect could be found (all p>.05).

Figure 4.6, left, shows the different layouts and their corresponding error rates. 2.8% of all authentication sessions using the telephone layout created input errors. The other layouts created slightly more errors, with the random layout performing worst (5% of all random layout sessions triggered an error). Even though the absolute numbers are low, this is an increase of 78.6% compared to the telephone layout.

Participants of the untrained group were responsible for 67.7% of all errors. More interestingly than that was the result that nine out of the 60 participants in the user study were responsible for 67.8% of all errors. That is, users that created errors were very likely to repeat them. Additionally, all critical errors happened within this group of nine. Seven out of these nine people were untrained users. This indicates that personal skills and training can be main factors for performance.

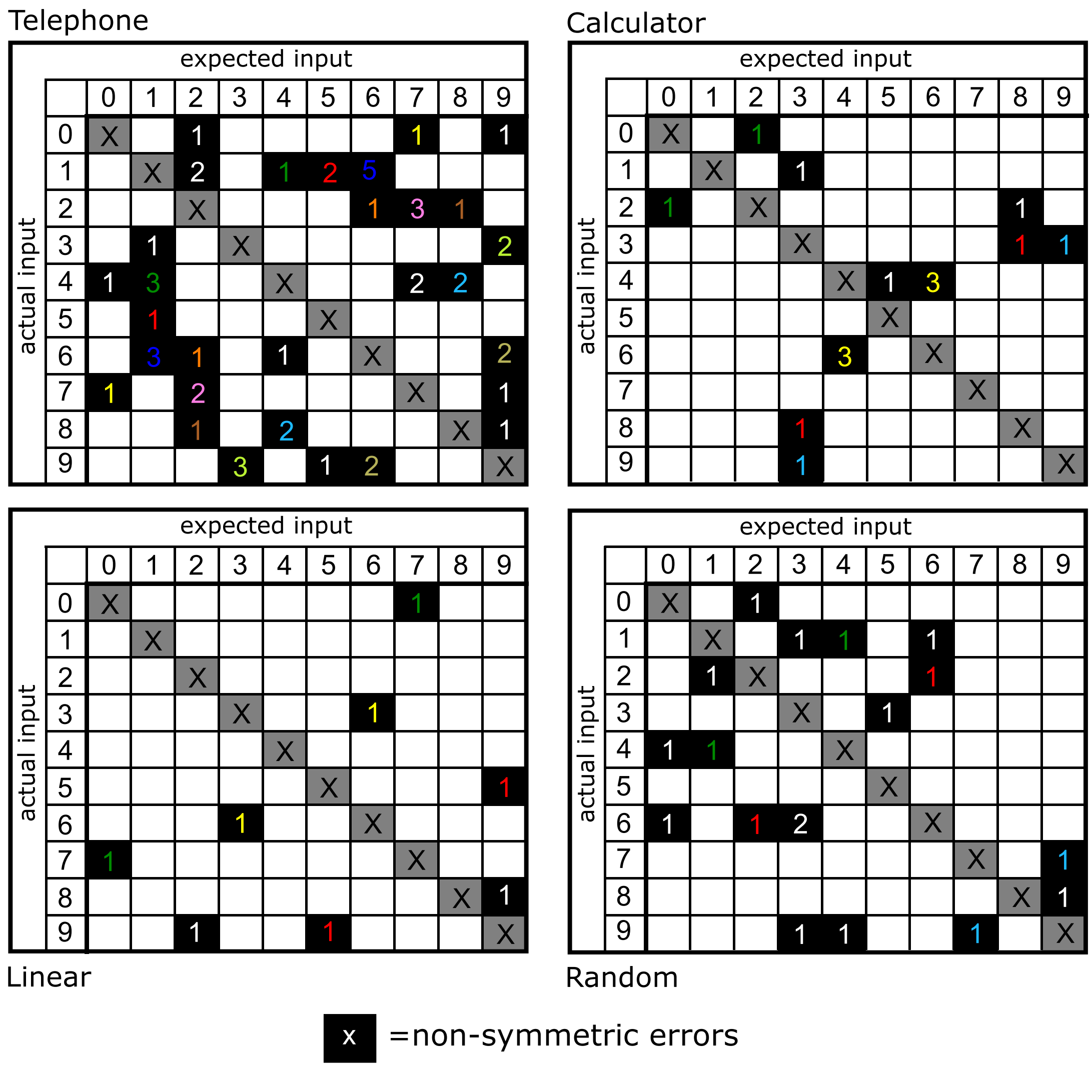
A performance analysis for the telephone layout over ten test days is depicted in figure 4.6, right. The graph shows that repeated use of the system reduces error rates. Consequently, no errors were performed on day 7, 9 and 10.

Having a look at the last six days of training phases of the main and the follow-up study, the observed error rates in both cases were lower than in the main part of the long-term study. 171 out of 174 (98.3%) of authentication sessions with the telephone layout were performed correctly. The three errors found did not contain critical errors. The 66 sessions with the random layout of the follow-up study did not lead to any errors. This indicates that introducing the consistent use of a keypad layout has a positive effect on error rates. Even providing a random layout, which by itself is not consistent, seems to show this effect. Nevertheless, lack of inner consistence of this layout has other drawbacks (e.g. on authentication speed) as previously discussed. Even though the quantitative results of the error rate analysis seem to support hypothesis (H2), due to lack of statistical proof, it could not be confirmed.

The subjective sensation of the participants revealed similar tendencies as the quantitative analysis. On a Likert scale from 1 (no risk for input errors) to 5 (high risk for input errors), participants rated the telephone layout best. 94.9% attested it to have no (76.3%) or only small risks for input errors (18.6%). The calculator (54.2% no risk, 16.9% small risk) and the linear layout (33.9% no risk, 39% small risk) reached similarly high ratings. The risk for the random layout was rated rather high (22% high risk, 20.3% risky). These results give additional support for the hypothesis that using alternative keyboard layouts has a negative influence on error rates.

4.3.3 Memorability

As mentioned before, each participant had the possibility to look up the PIN in the prototype system as many times as required. This way, it was theoretically possible for us to count the amount of forgotten PINs. Even though users were told to use this functionality, the overall number of look-ups was quite low. Only four participants actively used this functionality to look up their PINs for an overall of nine times. The final questionnaire as well as interviews with the



Telephone

expected input / actual input

	0	1	2	3	4	5	6	7	8	9
0	X		1					1		1
1		X	2		1	2	5			
2			X				1	3	1	
3		1		X						2
4	1	3			X			2	2	
5		1				X				
6		3	1		1		X			2
7	1		2					X		1
8			1		2				X	1
9				3		1	2			X

Calculator

expected input / actual input

	0	1	2	3	4	5	6	7	8	9
0	X		1							
1		X		1						
2	1		X						1	
3				X					1	1
4					X	1	3			
5						X				
6					3		X			
7								X		
8				1					X	
9				1						X

Linear

expected input / actual input

	0	1	2	3	4	5	6	7	8	9
0	X							1		
1		X								
2			X							
3				X			1			
4					X					
5						X				1
6				1			X			
7	1							X		
8									X	1
9			1			1				X

Random

expected input / actual input

	0	1	2	3	4	5	6	7	8	9
0	X		1							
1		X		1	1		1			
2		1	X				1			
3				X		1				
4	1	1			X					
5						X				
6	1		1	2			X			
7								X		1
8									X	1
9				1	1			1		X

x =non-symmetric errors

Figure 4.7: Error matrices for the different keypad layouts. The higher numbers of the telephone layout are based on the larger data set collected for it. Numbers indicate how many times a specific error was measured. Symmetric entries indicate that numbers have been mixed up during entry.

participants revealed several reasons for that. For instance, a user mentioned that she looked up the PIN in the original e-mail since she did not want to attract negative attention.

This is not a big issue for this study since we did not create it to analyze the memorability of PIN in general but the influence of different layouts on it. Looking up the PIN while the keypad is not visible does not provide any insights on this matter. The approach of choice for analyzing this influence was an in-depth analysis of error rates and corrections. That is, whenever an authentication session triggered an error (corrected or not), we interpreted possible reasons why this error occurred.

Errors that occurred during the study can be generally categorized as *memorability*, *oversight* and *operational* errors. Operational errors refer to usability problems like hitting the wrong button (e.g. among two neighboring buttons) or stopping the input too early. Oversight problems are based on the effect of change blindness [121]. That is, the participant did not realize that the layout changed and pressed the wrong button at the location where the digit was located in the original layout. This effect is only critical for the telephone, calculator and random layout having the same button arrangements. Memorability problems are identified based on the following criteria:

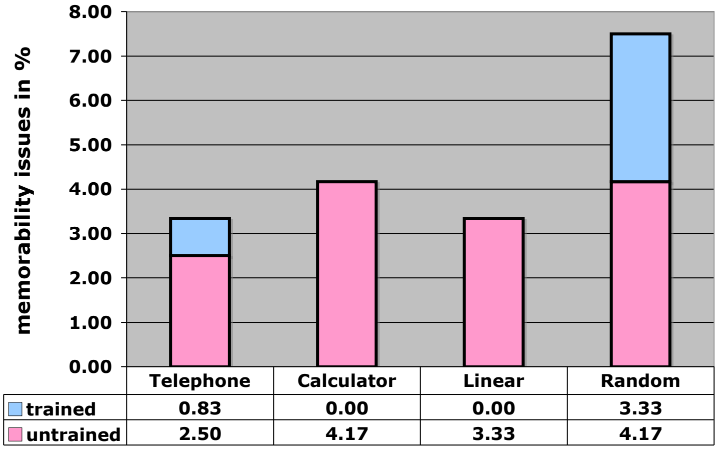
Repetition refers to participants repeating the same mistake in several authentication sessions. Such an error is in every case interpreted as having happened due to memorability problems.

Transposed digits occur when the input contains all the digits of the PIN but in the wrong order. Memorability issues of this kind are often boosted by language effects, when for instance, they are spoken in another order than they are written [95]. The English word “fifteen” (15) is a good example for this. All participants in the long-term study were native German speakers, a language in which this effect is even stronger. Transposed digits due to motor problems can be ruled out since the participants used pointing devices that only allowed for sequential input. Therefore, all transposed digit errors were accounted to memorability issues.

The distance of keys pressed can be a good indicator for memorability issues as well. If the space between the expected input and the actual button pressed is more than one key apart, the possibility of an operational error is very low and a memorability problem can be assumed.

Using this classification as indication for memorability problems is not perfect and there is a possibility for wrong interpretations or lost data. In some cases, information provided by the participants could be used to clarify specific findings. In the other cases, this classification is only the closest possible estimation.

Conrad et al. [21] used an analysis based on error matrices for their memorability evaluation. We used a similar technique as shown in figure 4.7. The limitation of this specific matrix visualization is that only completed errors can be visualized. That is, four digits (correct or wrong) have to be input to be able to show them in the matrix. Errors based on inputting too few digits are therefore



	Telephone	Calculator	Linear	Random
🟦 trained	0.83	0.00	0.00	3.33
🟪 untrained	2.50	4.17	3.33	4.17

Figure 4.8: Quantitative analysis of memorability issues sorted by the different layouts. The results show that with the exception of the random layout, untrained users performed worse than trained participants.

not shown. The degree of symmetry in the matrix indicates occurrences of transposed digits. The numbers indicate how often a specific error was repeated. With the exception of the random layout, the matrices can also be used to deduce the distances between the actual and the correct input.

14 out of the 21 incorrect authentication session with the telephone layout could be traced back to memorability issues. Six of them were transposed digits, five due to repetition and the final three based on the distance of the keys. Two memorability issues were additionally mentioned by the respective participants in the comment box of the study prototype. The telephone layout led to a big amount of repeated errors of transposed digits. However, the overall number of authentication sessions for this layout was much higher than for the alternative layouts. Therefore, this increased occurrence in the matrix is not a sign of stronger memorability problems of the design. Additionally, six out of ten corrected authentication sessions were due to memorability issues.

For the calculator layout, four out of six errors and one out of three corrections were due to memorability issues. One of the non-memorability errors was due to change blindness when a user input the digit 3 when the digit 9 was expected. This can be explained by the fact that in this layout, these two digits are swapped. For the linear layout, only three memorability errors were identified due to transposed digits. Finally, the random layout created the highest amount (relatively) of memorability problems. Seven out of nine errors were due to memorability issues. In one case, three out of the four digits of the PIN were wrong. The remaining six were due to transposed digits.

Overall, for the telephone and the linear layout, 3.3% of errors and corrections happened due to memorability issues. The calculator layout created 4.2% memorability related problems and the random layout had the highest possibility to create memory problems with 7.5%. The most interesting result is depicted in figure 4.8. It shows that the largest number of errors based on memorability issues was caused by the untrained group. Overall, 76.3% of these problems were caused by untrained users. Only the random layout is not easier for the trained group. Due to the overall low numbers of errors, no significant effects could be found. Therefore, hypothesis (H1) cannot be confirmed. Nevertheless, the trend indicates that the lack of conistency does have a bad influence on memorability.

This is supported by the results of the final questionnaire. Mostly trained users mentioned to have used visual or muscle memory strategies. The lack of the initial consistent training week of the untrained participants did not allow for developing such strategies. It is not surprising that on a Likert scale from 1 (no negative effect on memorability) to 5 (strong negative effect on memorability), the telephone layout was considered to have no (88.1%) or little (10.2%) influence on memorability. The linear layout (42.4% no effect, 16.9% little effect) was considered only slightly worse than the calculator layout (67.8% no effect, 22% little effect). This result partially contradicts the quantitative findings of the study. Even though the random layout was rated worst (16.9% strong effect, 15.3% high effect), a large group of 39% of participants attested it to have no negative effect on memorability. Interestingly, mainly trained users rated the negative effect as very high. This can be explained with the fact that their advanced learning strategies worked specifically bad using the random layout.

4.4 Discussion

The study proved two main assumptions. Firstly, consistency can be considered much more important for authentication mechanisms as commonly assumed. The results lead to the conclusion that randomization, as a common tool for providing security, comes with a pile of drawbacks that cannot be easily ignored. In the following, inner and outer consistency will be differentiated and their influence of authentication performance will be discussed. The second finding is on how the speed of authentication mechanisms is analyzed and evaluated. The study provided clear evidence that only putting such a high amount of effort on this task can reveal important findings.

4.4.1 Correctly Reporting Authentication Speed

As shown in the results section, correct time measurement makes a difference and can reveal issues that would stay covered otherwise. Applying vague (standard) methods for time measurement, one of the conclusions from the study would have most likely looked something like:

“The evaluation showed that even though random keypad layouts have a small disadvantage in authentication speed, the overall performance is comparable to that of a telephone layout.”

Applying in-depth time measurements, we could show that the telephone layout has a steep learning curve when it comes to preparation times, which random layouts do not have. This means that only the detailed analysis could reveal the fact that the telephone performed better than a random layout, which would have been overseen otherwise. Thus, the previous conclusion would have simply been wrong.

Based on this, we propose that whenever evaluating an authentication system, an approach similar to the one proposed in figure 4.2 should be used. This way, it is very unlikely to oversee important facts. Additionally, it resolves an issue that was discussed earlier in this work. It makes different authentication mechanisms more easily comparable to each other. Even if not analyzed, the different phases should at least be reported to enable others to adequately compare different authentication methods.

A big issue that has to be solved for each mechanism and setting is how to measure preparation time. Many systems do not have a dedicated starting point as the one we simulated in our experiment. Therefore, measuring this time is rather hard and requires a lot of creativity and extra work by the researcher. For instance, in another experiment, we designed and evaluated an authentication mechanism for a mobile device based on grasp recognition. To measure the starting point, we built a hardware device that enabled us to measure whenever the mobile device was lifted which allowed us to get a fix on the exact starting point of the authentication task and thus measure preparation time.

Additionally, some authentication mechanisms can require new preparation during the actual active authentication phase (e.g. if they are based on several consecutive challenges). Based on the results on the influence of preparation, this can be considered a drawback of the system and is very likely to make muscle memory strategies impossible. To some extent, randomization can thus be considered a usability problem of authentication mechanisms. This is more problematic since randomization is often used as a simple way to provide enhanced security.

4.4.2 Influences of Consistency

Related work already gave first indications that consistency is of high importance especially for a short and focused task such as authentication [107]. Till now, one of the main drivers for consistency has always been accessibility [58, 90, 105]. Within the long-term study, we found several results that support the assumption of the necessity of consistency, performance-wise as well as memorability-wise. Therefore, the results give further evidence that consistency is of high importance. Within this work, two types of consistency are distinguished. Firstly, *outer consistency* refers to the consistent use of the same authentication method. Strictly speaking, the different keypad layouts of this study could be considered different authentication mechanisms. *Inner consistency* on the other hand means that when using the same authentication mechanism for two different times, the user has to do the same input in the same way to authenticate. In the example of this experiment, inner consistency is therefore absent when using a random layout since the users have to press keys in a different order every time they want to authenticate. Talking about consistency within this thesis refers to inner consistency if not indicated otherwise.

The analysis of the different time measurements revealed that the results are influenced by extreme values. Especially preparation times are influenced by those values. Having a look at the 95th percentile of the results of the preparation times (thus ignoring the last 5% of times), shows that the maximum becomes the more extreme, the more the layouts differ from the participants' standard layout, the telephone layout. For the telephone layout, the maximum of the 95th percentile is 2.58 seconds. Using the calculator layout already increases this value by 45.3% (1.17 seconds). The linear layout increases it by 66.7% (1.72 seconds). The layout with the biggest difference from the telephone layout, the random layout, adds 214.3% (5.53 seconds).

This increase can be explained with human perception. Unfamiliar layouts have to be scanned sequentially to find the right digits. From study observations, we know that especially for PIN, participants tend to start the input after they successfully have identified all required digits. Another influencing factor is the fact that many participants used strategies based on visual and muscle memory cues (simply said, remembering the positions instead of the digits). These strategies do not work when the layout changes and thus the PIN has to be remembered in a more complex way. Thus, the lack of consistency has an additional negative influence if consistency is expected by the participants due to the normal authentication approach.

Results of the follow-up study using the random layout are better when it comes to preparation times and error rate. For instance, participants in the follow-up study created hardly any critical errors using the random layout while this layout caused most of the critical errors during the main study. This gives additional support to the assumption that consistency is an important criterion. Lacking inner consistency in the keypad layout (being random), the participants had to employ other (less effective) strategies to remember their PINs. Therefore, the preparation phase of the follow-up study only consists of the time required for the sequential search and advanced memory strategies could not lead to confusion due to their absence.

Based on these results only, one could assume that the lack of inner consistency in the random layout is acceptable since it forces the users to apply other strategies. However, in chapter 4.3.1, another advantage of the use of a consistent layout (outer consistency) was highlighted. As opposed to the random layout, using the telephone layout allowed for significant learning effects. That is, participants employed advanced strategies that they improved over time which resulted in a significant improvement of authentication performance. Additionally, the overhead created by the preparation phase of the random layout is nearly 100% as shown in chapter 4.3.1. Also the error rates caused by memorability issues did not improve by training when using the random layout as opposed to the other layouts as shown in figure 4.8. This can be attributed to the lack of inner consistency as well and is a problem for a time-sensitive task such as authentication.

Due to the lack of inner consistency, randomized systems do not cope (or work together) with the diverse strategies that users employ (like using shapes and the like). This was confirmed by the results of the final questionnaire. Especially trained users noted that they used visual or muscle memory learning strategies which are superior to plain memorizing. Thus, it has to be considered an important criterion of authentication mechanisms for public spaces to provide consistency (of some kind). This helps to improve both, performance and memorability due to muscle memory effects that can improve memorability even after a long period of non-use [51]. For PIN-entry, it is really helpful that positions can be easier remembered than distances [64].

Another interesting finding on consistency issues was on the performance of trained versus untrained users. Even though no significant differences between those two groups were found, the results of active authentication times show that especially for the random layout, trained users have a strong tendency to be slower. For the random layout they performed on average 0.41 seconds worse then untrained users. This can be explained by a higher degree of confusion caused by alternative layouts in the trained group compared to the untrained users and different memory strategies. The preparation speed of trained users on the other hand was faster for the telephone layout.

4.5 Lessons Learned

Only performing a long-term study like the one presented in this chapter helped to refine and specify two important criteria for the design of authentication mechanisms in general but specifically for public spaces: consistency and importance of in-depth time measurement.

Consistency is considered an important aspect in many areas of human-computer interaction, like in user interface design [112]. Even though sometimes seen as something negative [57], it has become a constant in software design. Talking about authentication mechanisms, consistency has never before been considered an overly important factor. Based on the results of the study, we argue that both inner and outer consistency are important factors for authentication mechanisms. Considering the fact that authentication is a time critical task that users never see as their primary goal (and simply want to get over with), it is even more important than for standard user interface design. A problematic finding was on inner consistency. If we consider randomization a bad design choice, the most common tool of choice for providing security has to be critically reassessed. For instance, ColorPIN [32], as presented in chapter 3.4.1, uses randomization in combination with limited inner consistency. This way, participants in the study were enabled to use more elaborate strategies to perform the authentication which led to a remarkable improvement of performance. The conclusion simply has to be that the higher the inner and outer consistency, the better the performance of the system.

A thorough approach for measuring time as the one proposed in this work (see figure 4.2) has never been proposed before. Even though similar data is often collected, it has never been reported or analyzed. Reasons might be that its importance remains mostly hidden. Only an in-depth analysis of data collected over a longer period of time allows for completely revealing its benefits. In chapter 3.3.2, MobilePIN [29] and its evaluation are described in detail. Realizing that the time needed to connect the mobile device to the public terminal and that this time has to be counted as part of the authentication time (despites cases in which the device has to be connected for the actual interaction) gave first indications that current time measurements had to be rethought of. In the scheme presented in this chapter, the time required for connection has to be categorized as preparation time. That is, an action that has to be performed before the actual authentication task can be started, but without starting it is not possible. The long-term study showed that precisely measuring authentication speed is also important for cases in which no obvious actions (like connection) take place and still provides important insights.

The long-term study on keypad layouts helped to refine these two criteria. However, both are highly technical. Consistency influences the (interface) design of an authentication mechanism already in an early stage. Authentication speed measurement on the other hand takes place in a later (or final) stage of the development process. The open gap that remains is on how “human setups” influence the design and evaluation of criteria in contrast to technical issues. Formulated as a question, this means: “What are behavioral criteria that influence the design of authentication mechanisms?” To fill this gap, we conducted a long-term field study on real use of public terminals (ATMs) which will be presented in the next chapter.

Chapter 5

Authentication in the Wild

Man is many things, but he is not rational.

– Oscar Wilde –

If we are not rational, then how can we expect from each other to behave securely? This is a very important question in general but especially when it comes to the security of computer systems. As a conclusion of this simple quote, researchers working on usable privacy and security have to deal with irrational users and irrational behavior. Education and training are often considered solutions for this problem and there is proof that they can sometimes work [79, 119]. A more promising approach, however, seems to be to solve these problems at a user interface level rather than shifting the responsibility to the users [111].

To find appropriate solutions for irrational behavior, it is important to know what exact behavior causes problems. Due to the sensitivity of such data, there is only little knowledge about behavioral factors that influence the design of secure authentication mechanisms for public spaces. Such factors can influence performance like authentication speed or acceptance, but they can also directly harm the security of proposed systems. Perceived levels of privacy, intimacy and security, time pressure and anxiety were previously identified as important factors influencing the decision whether or not to use an ATM [83, 84]. However, this data is usually based on different kinds of interviews and does not give insights on actual performance at and use of public terminals.

The approach taken in this chapter goes one step further. We performed a series of field observations of ATMs to explore how users actually interact with them and to find out about the influence on different behavioral factors [33]. Field studies, have the ability to uncover facts that would remain hidden otherwise (e.g. [63, 100]), for instance, since interview partners might not admit or think about them. The main focus of the observations was on the ATM authentication process, i.e., how people enter their PIN, whether and how people protect their PIN-entry from skimming attacks, and what contextual factors affect security and secure behavior.

After analyzing the first field study, we conducted two additional follow-up studies: A second field observation with the focus on obtaining more detailed interaction times, and an additional set of interviews in public spaces in order to ground some of our findings. All results of the studies point to one important conclusion: behavior is mostly irrational and seldom secure. Besides this, the studies gave deep insights on the following[1]:

a) The observations provided various insights on how users really interact with authentication mechanisms in public spaces. Especially findings on insecure behavior were completely unknown or unpublished before and quite surprising.

b) Based on these insights, a number of behavioral criteria were derived that directly affect the design and evaluation of authentication mechanisms for public spaces.

c) The study allowed us to derive several lessons learned and recommendations on how to perform field studies about the use of privacy and security relevant technologies. Details on these results can be found in [33].

5.1 Field Study Methodology

To optimize validity, the field observations were performed in six different locations in two central European cities, Munich (Germany) and Delft (the Netherlands). We chose ATMs that were available 24 hours a day, seven days a week. Due to legal issues, they had to be located outside. Additionally, this allowed for unobtrusively observing the actual ATM interactions. The observation method will be presented later.

The data of the primary field observation was collected over a period of nearly two months. The minimum number of visits per ATM was at least four times, with at least one observation session on a Sunday and at least one session during “rush hour” (i.e., mid-mornings, noon, or early evenings). This was necessary to ensure that the data collected was as broad as possible and did not, e.g., only include off-peak times, which could have biased the results. Rush hours and off-peak times were identified in pre-observations. Depending on the location (for instance, one was close to a supermarket) these times differed not only between cities, but also between locations within the cities. For instance, the rush hour close to a supermarket was between 5pm to 7pm while the rush hour at an ATM in a pedestrian area with shops and restaurants was during lunch time (around 1pm).

During the pre-observations, we noticed that terminal software can significantly differ from one bank to another and have influence on the performance. Therefore, we also made sure to observe a variety of ATMs from different banks (six banks in total) to avoid influences of the software

[1] We published the studies presented in this chapter in 2010 [33]. This chapter is partially based on this publication.

on the results. Each bank within the study used different software. At each ATM, 60 users were observed, resulting in an overall data set of 360 users, which were collected during 44 observation sessions. 199 of the observed users were male, 161 female.

All observations were performed and recorded by the one and the same researcher. This was necessary to keep the data comparable, since different people might apply different standards during the observation, deliberately or not. Even though multiple observers might have reduced the risk of accidentally missing data, we opted for this solution since we rated consistency over efficiency (speed of collecting the data).

The most undesirable influence on the data would have been if users would have realized that they are being observed. Therefore, in order to remain unobtrusive during observations, we chose ATMs that were visible from public outdoor seating areas, i.e., street cafés and restaurants that had tables in appropriate positions outside. Surprisingly, a large number of outdoor ATMs that we found were actually close to such spots. Thus, finding appropriate locations was not an issue. Considering these precautions, it is very unlikely that the observer did arouse suspicion amongst ATM users. Additionally, the single observation sessions were kept rather short to minimize this risk.

5.1.1 Ethical and Legal Considerations

Whenever users authenticate to an ATM, the data they are using is highly private and confidential. Therefore, in order to ensure the privacy of the study subjects, we chose all observation spots in such a way that the hands of the subject could be seen but the keypad itself was not visible. This allowed for the required observations but prevented the observer from being able to spy on the users' PINs. Also, the observer was positioned at a distance where the ATM screen could not be read. Most importantly, all observations are based on written data by the observer – no surveillance technology of any kind was used, i.e., neither videos nor photos were made.

We instead used a written checklist in order to ensure that no important information was missing. This list was based on procedures identified during an informal pre-study and highly optimized. The checklist included the following information:

- location (of the ATM)
- gender
- time of day
- interaction time (for how long the user occupied the ATM)
- queue length behind user
- security measures

- start of interaction
- repeated PIN-entry (yes or no)
- comments

In the primary field study, *interaction time* was simply measured with a standard commercial stop watch. Measurement started the moment a user inserting the bank card. The time was stopped when the user took the withdrawn money (all our observed interactions resulted in a money withdrawal). We later performed a more detailed analysis of interaction times in a follow-up study (see chapter 5.1.4). The entry *security measures* featured a number of checkboxes for marking procedures that had been identified in the pre-study, such as "hiding PIN-entry with other hand" or "checking people standing close to the ATM". Finally, situational information that could not be narrowed down to a set of actions was written down in the *comments* section of the checklist. Such information included "in company", "on the phone" or "carrying shopping bags".

To ensure untainted data, observations were only added to the data set if all of the points in the checklist could be identified and collected with 100% confidence by the observer. The reasons for failed observations were mainly cars or other people that suddenly blocked the view to the ATM or the user. Roughly one third of all observations were thus discarded. There were some rare instances of interesting behavior (e.g. a user leaving the ATM after a failed authentication attempt) that led to failed observations – these were also not added to the data set, but instead written down as additional comments in case they would help to gain further insights.

In the countries where we conducted the studies, no ethical review boards are in place for this kind of research. However, legal issues have to be considered. For instance, German privacy regulations state that without the explicit consent from the subjects, data can only be collected and stored anonymously[2]. However, once data has been rendered anonymous, it can then be used freely for scientific purposes. In addition, it can even be shared amongst scientific institutions. Since none of our subjects can be identified by any means (no videos and photos were taken), our data collection is truly anonymous. Furthermore, as the study was conducted in public spaces without the use of AV-equipment, our local legal counsel informed us that no consent from any institution (e.g., banks or city administration) was required. In connection with the previously mentioned measures to protect the subjects' privacy (e.g., not being able to see the actual PIN entered), we thus did not identify any legal or ethical issues with this study.

5.1.2 Methodology Limitations

Since ATM interaction is a sensitive and private task, it was very important for us not to disturb the users or to negatively affect their privacy. Therefore, we decided *not* to engage them in interviews after the observation. Consequently, some of our findings are necessarily based on (speculative) reasoning about the observed behavior, rather than on actual user feedback. Especially inferences

[2] Exceptions do exist of course, e.g., for law enforcement or the protection of private property.

on the use of security, the influence of other people (company), and queuing strategies were not verified with those users exhibiting such behavior. To fill these gaps, we performed additional interviews in public spaces with a focus on these aspects (as will be described in chapter 5.1.3).

When analyzing the observational data from our first study – and especially the comments – it became apparent that the time measured from entering the ATM card to the moment of money withdrawal was not entirely sufficient for analysis. Many users blocked the ATM for a significantly longer amount of time before and after the actual cash withdrawal, which we referred to as *preparation phase* and *cleanup phase*, respectively[3]. These phases include simple tasks like getting the ATM card from the wallet or putting down shopping bags. Based on our experiences from the first study, we reckoned that this overhead might in some cases be around 50% to 100% of the "interaction times" that we measured. To clarify this issue, we performed a second set of observations with a focus on measuring interaction times.

5.1.3 Follow-Up Study: Public Interviews

To get a better understanding about the observed behavior during the primary study, on users' security considerations, the influence of company, and users' queuing behavior, we conducted a number of public interviews some time after our initial field study. Interviews took place over a period of one day in the city center of Lugano (in the Italian speaking part of Switzerland). As we did not want to interview people who we had previously observed withdrawing money, we do not think that the change of location for these interviews affected our findings. Also note that these interviews did not attempt to achieve statistical significance. We merely wanted to gain some insight into "people's thinking" with respect to ATM usage. While there might clearly be cultural differences between ATM users in Munich, Delft, and Lugano, we expect to be able to uncover the same basic set of attitudes in each of these locations (though we do not have evidence for this assumption).

Overall, 25 full interviews were conducted. That is, 25 participants answered all questions. Additionally, two interview partners stated to never use ATMs and thus were not asked any additional questions. The average age of the survey participant was 36 years. The youngest was 19 and the (two) oldest 64 years old. One participant did not agree to share his birth year with us. 16 participants were male, nine were female.

Two interviewers performed the interviews together. They were fluent in English, German, and Italian, and thus were able to cover a large range of possible interview partners. While we did not record nationality, all interviewees were in fact fluent in at least on of those three languages. People were semi-randomly picked. "Semi" refers to the fact that the interviewers tried to get people from as many different age groups as possible. Firstly, people were asked whether they would be available for a short interview. They were told that the interview was for a research

[3] Those phases are named in the same way, as the in-depth time measurement presented in chapter 4.2.2. However, they do not refer to the exact same conditions but rather to tasks before and after ATM use. Based on their similarity, we decided to stick to those names introduced earlier.

project of the local university and that no private data of them would be collected. Approximately 30% of the approached people did not agree to participate in the interview.

The first question was about whether the interviewee actually used ATMs or not. Out of 27 interviewees, only two stated that they did not use ATMs at all. One person explained this fact with: "*I don't trust those machines, so I don't use them*". For participants who said that they used ATMs, we continued with the following questions:

- Approximately how many times per week do you use ATMs?
- Do you worry that someone might steal your PIN when using an ATM?
- How do you protect your PIN-entry?
- If you are in company, would you still protect your PIN? (If no, why not?)
- If there is a queue at the ATM, would you wait in line? On what does your decision depend?
- What is your alternative to queuing at an ATM?

Participants were told that they could answer those questions freely. We did not interrupt them as long as they felt like talking. During that period, the interviewers took notes to record the answers – again, no recording devices other than pen and paper were used. After each interview, participants were given a small reward (sweets) for answering the questions. The final question asked whether they were willing to provide us with their birth year. All but one participant gave us this information.

5.1.4 Follow-Up Study: In-Depth Time Measures

One of the main limitations of the primary study was that it did not allow for in-depth insights on how much time different phases of interaction took the users. Additionally, time overhead that is spent at ATMs besides the previously measured "interaction time" could not be identified. Therefore, we conducted a follow-up field study in Munich (Germany). In contrast to the first study, we used a custom software installed on an Android-based smartphone to easily record the individual interaction phases. Meant as a supportive study to gain insight on the influence of preparation and cleanup on the overall interaction time, this study featured a smaller amount of only 24 observations. The in-depth measurements were performed on two ATMs that were also used in the primary field study. At each ATM, twelve data sets were collected in four observation sessions. Three different times were measured as shown in figure 5.1.

Preparation is defined as the time from the moment the ATM is blocked by the user to the beginning of the actual interaction (i.e., the previous begin of the "interaction time" of the main study, when the user entered the card). Blocking was defined as when other users were not able to use the ATM.

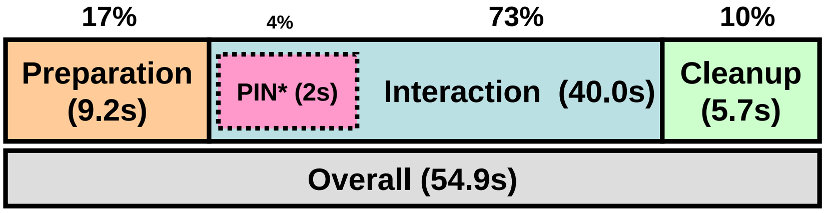


Figure 5.1: The different interaction phases measured during the in-depth time study with their average times. **PIN was not measured and is based on results from lab studies*

Interaction refers to the previously measured time starting from card entry until cash/receipt withdrawal (previously called “interaction time”).

Cleanup describes the additional time the ATM was blocked by the customer after the interaction phase. Actions performed while still blocking the ATM were for instance putting the money or the cash card back to the wallet.

Splitting an interaction up into several consecutive steps can help to identify usability factors and to uncover different effects that might have stayed hidden otherwise. This was for instance shown by Bauer et al. [5] when they analyzed the usability of the Grey authentication system.

Apart from the use of a smartphone to acquire more in-depth recordings of times, the same methodology and ethical rules were applied for this study as they were for the initial observations. Thus, due to the private nature of the observations, we could not record more detailed breakdowns of the interaction time, in particular the time spent for entering the PIN. This would have required us to observe the actual ATM screen, which we tried to avoid for ethical reasons. Based on previous work, however, we know that PIN-entry is very fast and usually takes around two seconds only (e.g. [34, 27]). We tacitly assumed similar timings for our observations.

5.2 Findings

This section presents findings based on the two field studies and the field interviews, grouped along five main properties: overall interaction time, user distractions, input errors, queuing behavior, and employed security measures.

5.2.1 Interaction Time

In the main field study, an interaction session took on average 45.9 seconds (SD: 15.1s). The fastest user finished in 19.9 seconds while the longest took 125.3 seconds. Sessions were typically measured from the moment the user inserted the card until the cash or the receipt (if any) was taken. As we pointed out above, our observation positions did not allow us to isolate authentication times (i.e., PIN-entry) in these measurements – taking PIN-entry measurements from prior work [34, 27] (2 seconds) these would thus be less than 10% of the total average interaction time that we observed.

A detailed analysis of the data revealed that factors like queues and the use of security measures did not significantly influence interaction time. For instance, people hiding their PIN-entry (mean: 45.9s) did not take significantly longer than users that did not perform such security measures (mean: 44.4s).

However, during our observations we noticed that the actual interaction with the ATM was only part of the time that a single user blocked the machine. Significant overhead came from "preparation" and "cleanup" actions taking place before and after actual ATM use, respectively. These actions included: arranging shopping bags, finding the bank card, putting the withdrawn money into the wallet, arranging personal items (e.g., putting away the wallet), and finishing a phone call or a conversation with a friend.

These times were measured in the follow-up study described in chapter 5.1.4. The in-depth measurement later showed that preparation and cleanup actions would take around 27% of the time that the ATM was blocked (17% preparation, 10% cleanup). In one extreme case, preparation and cleanup made up 66% – for a user that arrived with a dog and a child in a pram. Before he could use the ATM he had to make sure they were safe (e.g. blocking the wheels of the pram), which he later had to undo again during the cleanup phase. In the "best" case, they took only 16% of the time that the ATM was blocked. This was achieved by a user that performed a strategy that we could observe four times during our observations: he had his cash card already prepared when he approached the ATM, rendering the "preparation" time practically zero.

The average time for preparation (9.2s) was higher than for cleanup (5.7s). The different phases, their average times, and percentages are depicted in figure 5.1. The most important thing to notice here is that standard PIN authentication takes only a fraction of the overall time that a user is in front of an ATM.

5.2.2 Distractions

The primary study revealed a number of factors that distracted the user during the actual ATM interaction. They either interrupted the interaction with the ATM or slowed down the preparation or cleanup phase. One of the most common distractions were friends or partners that spoke to the user during the interaction. Other factors were for instance shopping bags or prams and their content that partially required the continuous attention of the user. Overall, 38 people (11% of overall users) were distracted by various factors during their ATM use as shown in figure 5.2.

In an extreme case, a user came to the ATM with a dog and his child in a pram. Before he could even think about starting the interaction with the ATM, he had to take care of both, effectively blocking the ATM in the process. Also during the interaction, the child repeatedly required attention, resulting in a loss of focus on the actual task.

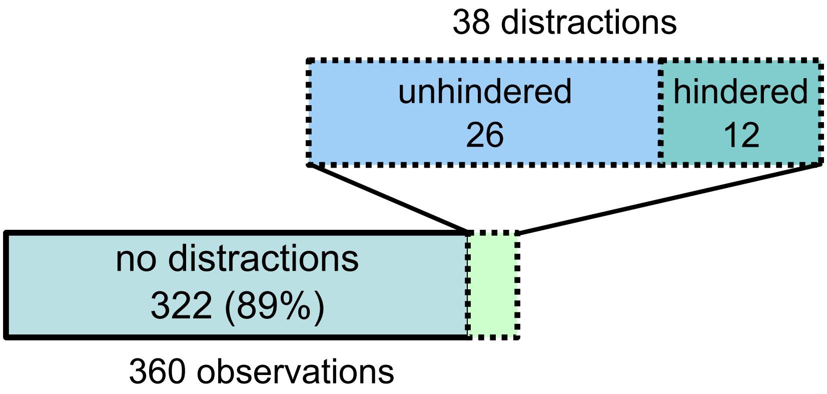


Figure 5.2: Distractions that occurred during the ATM field study. 11% of users were distracted during PIN-entry, for 3% this even hindered PIN-entry and led to errors.

5.2.3 Input Errors

ATMs, as most public terminals, stop authentication attempts when a critical error occurs. That is, they give users three tries to authenticate to the system. In case the user fails to do so, the bank card will typically be confiscated by the machine. While the distance to the ATMs did allow the observer to see only the general interaction with the keypad, but not the actual PIN-entry, we distinguished errors from successful input in the following way: All ATMs in this study used screen keys for providing access to their services (e.g., account balance, withdrawal). To activate any ATM functionality after a successful authentication, the user had to use one of the screen keys. That is, the hand had to be moved away from the keypad to the screen. Going directly back to keypad input without touching any of the screen keys thus meant that the user was forced to correct the PIN. In some of the cases, users even removed the card after an error occurred and restarted the authentication process.

Out of the 360 users we observed in the initial field study, only six failed to authenticate correctly at the first attempt. These six users subsequently spent more time ensuring that they would "get it right" on their second attempt. The time for an interaction that included a failed authentication session was 103.1 seconds on average – more than twice the average time of a session without failed authentication. Due to the small amount of errors, this difference is not statistically significant. However, the low error rate correlates with standard PIN-entry error rates that we know from laboratory studies (e.g. [34, 106, 110]).

We observed one user who at first applied security measures but failed to authenticate correctly: shielding her PIN-entry with the other hand meant that she could not see which buttons she was pressing. After her first attempt failed, she gave up on shielding her PIN-entry and then was able to enter the PIN correctly.

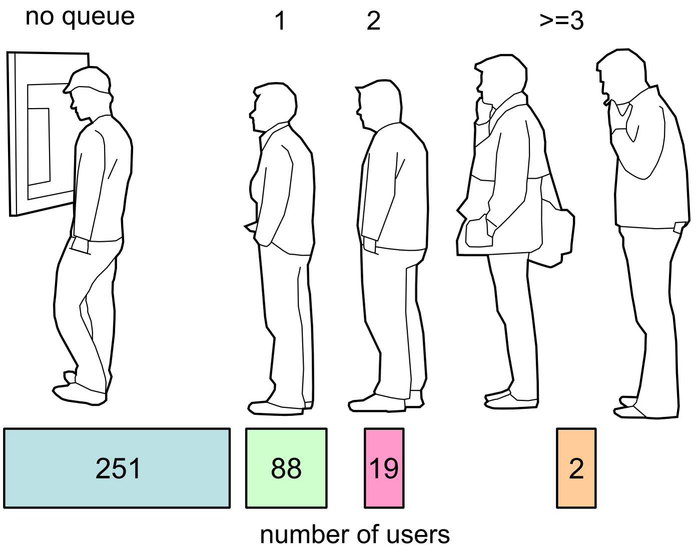


Figure 5.3: Different length of queues as observed during the ATM field study. In 251 cases, no one was queuing behind the current user. Only in two cases the queue was longer than two people.

5.2.4 Queuing Behavior

If an alternative authentication method takes longer than PIN-entry, one might expect this to have an effect on accumulated waiting times. If authentication took, say, twice as long, would queues in front of ATMs get much longer? During our observations, we were thus interested in actual queuing behavior: how long do ATM queues typically get, and how do people deal with long queues, both while waiting and while withdrawing.

In the following, we count only people in line to use the ATM, not the people accompanying them. Big queues almost never occurred during our observation sessions as shown in figure 5.3. In 251 of the 360 sessions, no one was queuing behind the user. Queues with a length of one appeared 88 times, queues with a length of two 19 times. We only observed two instances when the queue had three or more people: one time three people where queuing, once we saw four people in line. At a length of two, we saw people approaching the ATM but when they realized there was already a queue they seemed to change their mind and turned to go away.

To get a better understanding of this behavior, and to understand reasons for and against queuing, we included two corresponding questions in our follow-up interview study. When asked if they would queue in front of an ATM, three of the 25 interviewed participants stated that they would never queue. Four users said that they always queue, no matter how long the queue. The remaining 18 participants stated that it would depend on the circumstances. When analyzing the interview logs, we identified four such influencing factors: *Urgency*, *queue length*, the *availability of alternatives*, and the *perceived safety* of the queue. We will briefly describe these factors in turn below. Note that Little et al. [83] also identified *time pressure* as an important factor toward

ATM use. However, this was only mentioned by one of the 25 participants. We assume that our way of phrasing the questions for the interview did play a role in this notable absence from the list of factors.

Urgency 11 out of the 25 interviewees were only willing to queue if they urgently needed cash.

Queue length Six participants explicitly mentioned a queue length that they would consider acceptable. None of them said they would accept a queue bigger than three. One user said that he would only queue if it was urgent, and only if the queue length would be two at maximum.

Alternatives The most important factor for the participants when deciding on queuing was the availability of alternatives – not only the alternative of having another ATM close-by, but also other means.

14 participants stated that they would only go to another ATM if a) the alternative ATM would not apply charges, and b) if it would be located close-by. Two participants mentioned that they would always queue due to the lack of alternatives: both were with banks that had very few ATMs in town from which they could withdraw money without being charged. For instance, one user stated that he would always queue, since there were only two ATMs in town that he could use without being charged, and these were located at exactly the opposite sides of the city. Another user said that his bank would not provide many cash machines and thus he is usually bound to using the one that is available at the moment, which means he would queue. Four users stated that a queue would make them skip cash withdrawal altogether, given that they were on their way to shop at a place that supported paying by card (e.g., a local supermarket).

Perceived safety One participant had a different view on ATM queuing than the rest of the interviewees. Instead of considering it a time burden to queue, she instead considered the safety aspects of the queue. Depending on the type of people in line, she stated that she would not queue "*if there are strange people nearby*".

5.2.5 Observable Security Measures

During the primary study, we found that only 124 out of 360 users (around 35%) made observable efforts to secure their PIN-entry. 57 of those users were female, 67 were male. A summary of secured and unsecured input is depicted in figure 5.4.

The most common security measure was hiding the PIN-entry with the second hand or the wallet (120 out of 124). Many ATM interfaces propose this method in visual or textual form when prompting for PIN-entry. Figure 5.5 shows two mockups of real world examples of possible

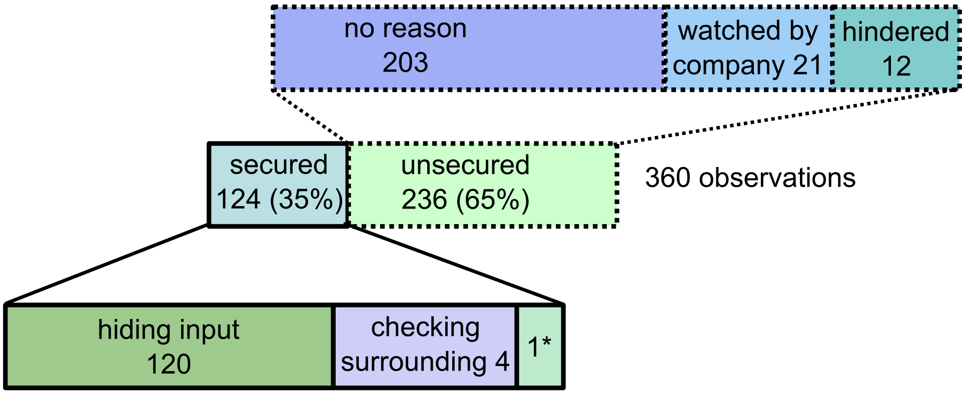


Figure 5.4: Number of users that did or did not apply observable security measures. *One user applied two different security measures including checking the ATM for manipulations.

notifications. Four out of the six ATMs in our study displayed such a hint. Interestingly, users at such ATMs were not more likely to protect their PIN-entry.

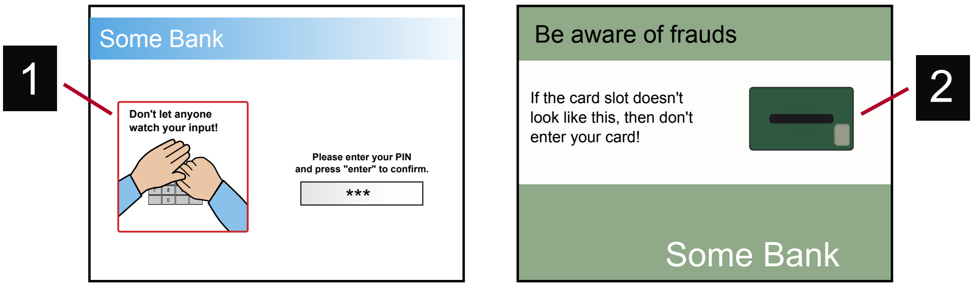
The remaining four users that applied security measures did not hide the PIN-entry, but instead checked their surrounding and verified that no one was standing nearby. One user additionally checked the ATM intensively for manipulations. To do so, he employed behavior as commonly proposed in the media and displayed on many cash machines. This mainly consisted of grabbing and shaking the card slot and keypad to look for loose parts.

With 236 out of 360, almost two thirds of the observed users did not observably secure their input in any obvious way. This number increases when considering the users that only weakly secured their PIN-entry. For instance, 15 users shielded their input only toward the screen, but left their PIN-entry visible from the sides.

In the interviews, we wanted to get a better understanding about reasons why users would not protect their PIN-entry. Therefore, we firstly asked them whether they were worried about someone stealing their PIN while using an ATM. 14 users, i.e. more than 50%, were not afraid of the risk of PIN theft. One of them even mentioned that "*the bank puts up cameras, so I am safe*".

Surprisingly, 19 out of the 25 interviewees (including some that said that they were not worried about their PIN being stolen) stated that they would actually take security precautions, with 11 of these mentioning that they would always hide their input. This is a much higher percentage than we found in our primary field study, where barely a third secured their input. While part of this discrepancy could be attributed to "white lies" during the interview, a closer look at our interview logs revealed a more nuanced explanation: Several of the mechanisms people said they employed to secure their PIN-entry were difficult – if not impossible – to detect during our observations. Consequently, the percentage of people securing their PIN-entry could have been much higher than 34%.

For instance, three participants mentioned that they would hide their PIN-entry with their body, blocking the view for onlookers. This is a rather large number considering that there was only



Some Bank

Don't let anyone watch your input!

Please enter your PIN and press "enter" to confirm.

***

Be aware of frauds

If the card slot doesn't look like this, then don't enter your card!

Some Bank

Figure 5.5: Examples of how ATMs visualize to their users that they should apply security measures. 1. Instructions to hide the PIN-entry. 2. A visualization of how a card slot that is not manipulated should look like. These screens are based on existing ATM interfaces.

a sample of 25 persons. However, during the field study, there was no situation in which a user efficiently blocked the view with his or her body. In all cases, our view to the keypad remained unblocked. Another three said they usually tried to choose ATMs inside buildings, or that they would always choose the same ATM as a security measure. Six participants mentioned that they would check the surrounding while they were approaching an ATM. If there was no one in sight, they would not hide their input. Since queues were rather seldom during our field studies, some users might not have hidden their input due to that reason. Finally, one user said that he would always do the input very quickly so no one could see it.

The majority of participants in the interview did not consider the danger of hardware based attacks, such as video recording and fake keypads. That is, many of the described measures – like fast input or hiding the input with the body – are rendered useless by those attacks. Therefore, a user might feel secure (e.g. when there is no one around) when she actually is not secure at all.

From both, our observations and the interviews, we can infer that many users do not protect their input (236 or 65% during the observations) – or do so rather ineffectively. However, the reasons can be manifold. Apart from the obvious lack of interest, or a lack of threat awareness, we found three instances in which other factors hindered PIN security: *physical hindrance*, *memorability*, and *trust display*.

Physical Hindrance Securing PIN-entry against cameras and shoulder surfers typically requires a second hand to shield the keypad. We observed several instances where users simply did not have a free hand to spare to protect their input. For instance, they were holding shopping bags that they did not want to (or were unable to) put down. Other users were holding their mobile phone, having calls or even holding children in their arms. Overall, twelve instances of hindered, unsecured PIN-entry were observed as shown in figure 5.4. An example of this (staged by the authors) is depicted in figure 5.6. From the data, we cannot infer whether hindered users would have secured their PIN-entry if they were able to.

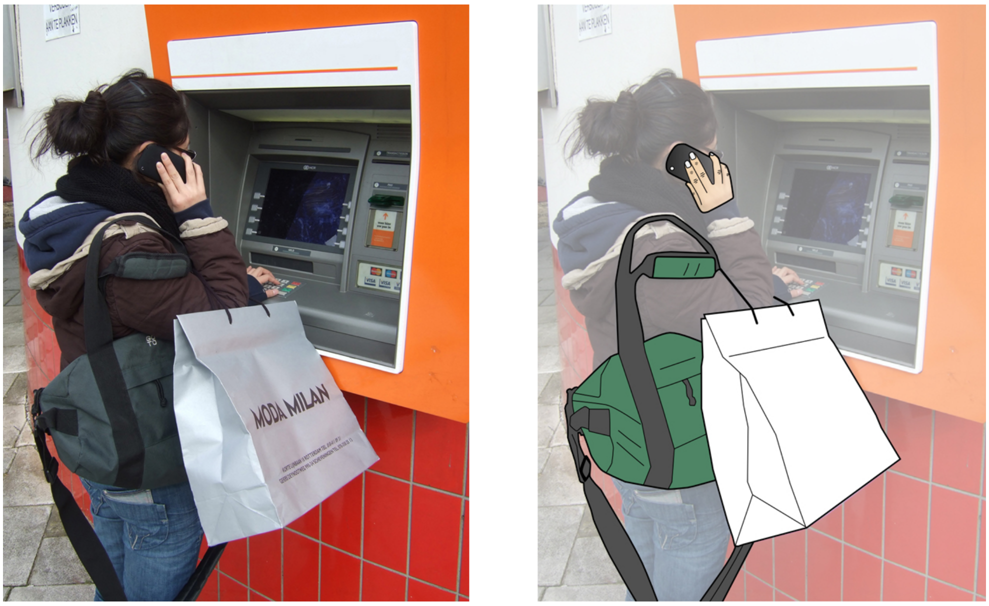


Figure 5.6: Physical hindrance can cause security problems when using an ATM. This staged figure shows a user who is on the phone, and is additionally hindered by bags and thus cannot protect PIN-entry.

Memorability Even though a four-digit PIN is a rather short token to memorize, the increasing number of cards and services that depend on different PINs can make it difficult to remember them, prompting research into more memorable authentication methods (e.g. [96]). While during the 360 observations we only observed four sessions in which users forgot their PIN, these four cases vividly document how badly PIN-entry fails when it does. Even though the first two cases were observed at two different ATMs on two different days, both users reacted in exactly the same way: after their first failed input attempt, both pulled out a notebook or piece of paper from their purses (in which they also kept their ATM card!) and consulted it for their PIN. After checking their notes in this way, both users could authenticate successfully. The third and fourth case showed similar behavior. Instead of having the PIN written down, however, those two users checked their iPods for their PINs.

Writing down PINs or passwords to remember them was already reported as a major problem of token based authentication systems (e.g. in [2]). Within the scope of authentication in public spaces, the danger increases since an attacker can even more easily get into possession of the token, which the user carries around.

Trust Display In many cases, users were with friends, family members, or partners as shown in figure 5.7. Of the 60 users that were not alone at the ATM, we found 22 instances (37%) in which users performed their PIN-entry in plain view of their company. "Plain view" not only refers to not actively hiding the input, but more often meant that from their position, the accompanying persons could easily gaze on the whole interaction. In one case, a father even dictated his PIN to his (young) son so that he could have the "fun" of entering it.

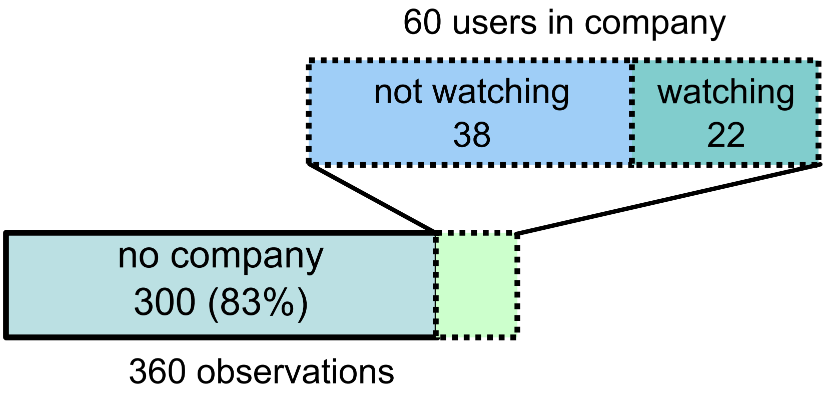


Figure 5.7: Number of users that were in company during the ATM study. 7% of users were in company, 6% let their companions watch their PIN-entry. *Only one user that was watched by her companions applied security measures.

Sharing (or at least not hiding) one's PIN in these situations might constitute a proof of confidence – or the other way around: hiding one's PIN might be constructed as a sign of mistrust toward the accompanying friends and family. The problem of social pressure and social factors has also been discussed by Kim et al. [74]. Social factors were one of their design criteria for their tabletop authentication system. To take the social pressure from the users, their systems are designed in a way that security is enforced and does not rely on the user as proposed in chapter 3.3.3. Our observations support the importance of social factors on security.

To get a deeper understanding on this, the last block of questions in the follow-up interview study was "*whether users would protect their input if they are in company*". 13 Participants stated that they would still protect it while in company. One of these 13 mentioned that whenever he is around friends that used an ATM, he would look away since "*I don't want to put pressure on them*". The remaining twelve said that they would not protect their input while in company. However, only four of them were users that stated to hide their input in general. Out of the participants that stated that they would not protect the input when friends were close, four stated that they would not protect it since they trusted their friends.

5.3 Implications

The insights we gained during our observation provided important feedback for the evaluation of authentication mechanisms for public spaces. The implications in this chapter are directly derived from our observations, support findings from other chapters and provide further insights into them, and finally describe new findings as well.

5.3.1 Authentication only a Minor Task

The numbers from our observations suggest that authentication only takes a marginal part of the whole interaction time with an ATM. With 46 seconds on average (or 54.9s when considering preparation and cleanup), more than 90% of ATM interaction is spent navigating menus and waiting for the withdrawn money and optional receipts to appear, etc. Distractions such as minding bags or talking to friends add further delay.

Seeing it as what it is, a minor task that has to be done to be able to perform the actual task (e.g., withdraw money), it is questionable whether significantly slower authentication systems will be accepted by the users. Considering an interaction time of 52.9 seconds, a system that takes, say around 12 seconds (e.g. [59]) adds an overhead of around 18% to the overall time.

The fact that we rarely observed longer queues (>2) during the observation, and that in the interviews we found that people based their decisions to queue or not on manifold factors, minimizes the “threat” of accumulated waiting times. We can therefore support survey findings from [83] that people judge waiting time with respect to time constraints and need for cash. It seems that a queue length of two is a borderline that many people are only willing to cross if it is urgent and if their time constraints allow for it. However, increased authentication time can also have an influence on people waiting in the queue and would increase overall waiting times over accumulation.

Considering common authentication mechanisms from literature like [17, 32, 54, 59, 106, 135], both waiting and overall interaction times can increase drastically if the authentication mechanism takes significantly more time. If, for instance, the interaction time for an authentication mechanism takes around 45 seconds (which is the average overall interaction time that was observed during the field study), the second user in the queue would have to wait twice as long as when using standard PIN-entry. This indicates that time is a very important factor when creating an authentication mechanism for public terminals, which can decide over acceptance or rejection of a system. It also supports findings presented in chapter 4 that authentication speed means more than just summing up times required for button presses.

Within this work, we cannot provide an exact borderline on how long an authentication mechanism for ATMs should be, and we argue that such a value does not exist. However, we expect that PIN authentication is only accepted by users since it is very easy and – maybe most importantly – extremely fast. Therefore, it is highly appropriate for authentication in public spaces, since the overall task is very short and PIN still only requires a small fraction of the overall time. A rule of thumb could be that an alternative authentication mechanism for public terminals should only require a fraction of the overall time (< 10%) that a user spends at the machine.

5.3.2 Memorability Issues

Out of the 360 users, only four were not able to correctly recall their PIN at the first try. While it could thus be argued that memorability is not a problem for the large majority of users, this might be premature. Firstly, in the few cases where it was a problem, severe security problems resulted (e.g. PIN written down). Secondly, our results are most likely biased toward the most often used PIN. If we would have required people to recall PINs of membership cards or seldom used credit cards (which increasingly require a PIN as well), we might have gotten a very different picture. Therefore, especially for authentication systems for public spaces, memorability deserves a lot of attention.

5.3.3 Security Should not Require an Active User

One of the most important findings of the work on VibraPass, as presented in chapter 3.3.3, was on bad lies. They indicated that the security (or the decision to apply secure behavior or not) should not be the burden of the users but something built-into the authentication mechanism.

There were several findings of this field study that support this notion. The security of an authentication mechanism should not rely on the way the user interacts with it. In some cases, physical constraints (e.g. heavy shopping bags) did not allow the user to apply additional security precautions since there was the possibility to skip secure behavior. Other examples had users try to hide their PIN-entry, but using an angle that left the keypad in plain view for a shoulder surfer. Much more often, however, it was the case that users did not even try to hide their input, either out of negligence or (potentially) as some sort of proof of trust.

Clearly, an alternative authentication mechanism needs to minimize the ability of the user to disclose the shared secret token by accident or through negligence. For instance, Sasamoto et al. [110] created a system that does not disclose the authentication token by simple observation. Also Kim et al. [74] created their authentication systems in a way that makes it impossible for the user not to hide it.

In other work it has already been noted that security is seldom the user's primary goal [65, 133] and that users are "bad" in protecting their authentication tokens [2]. These results support our claim for authentication mechanisms that have security built-in. However, this often comes at the cost of usability and has to be handled carefully.

5.3.4 Social Compatibility

When designing an authentication mechanism that does not require an active user, the problem of social compatibility might – but does not necessarily have to – already been solved. Results from the field observations as well as results from the field interviews indicate that social factors can lead to insecure behavior. Therefore, authentication mechanisms should be compatible with social norms. That is, to commit secure behavior, a user should not have to perform an action that might be misinterpreted as showing mistrust to a person accompanying her.

5.3.5 Authentication in Highly Distractive Environments

The observations showed that distractions can appear in manifold ways, and in particular in the form of ongoing social interactions like chatting. Authentication mechanisms for public spaces should therefore have a simple design and work even without full attention given to them. For instance, a fictitious authentication mechanism that requires the user to follow a row of events to get the authentication right, can easily fail if during one of the steps a distraction occurs. Therefore, it has to be possible for a user to recover from a distraction without having to start the authentication process over again.

5.4 Limitations of the Results

Since the main observation took place in two central European cities, it has only limited validity with respect to other cultural areas (e.g., Asia) or in less urban settings. For instance, it can be assumed that queuing and factors for it are different due to other cultural backgrounds.

The unobtrusive nature of the observations did not allow for in-depth findings on whether people check the hardware of an ATM (keypad or card slot) for manipulations. However, our general findings suggest that people only rarely use this security measure.

As for any study that involves direct contact to the participants, the field interviews might have been slightly biased since the participants might have wanted to "look good" or "do it right". Therefore, the numbers on secured input might be higher than they are in reality, which our field observations seem to confirm.

5.5 Lessons Learned

The main motivation to conduct real world observations of ATM use was to get insights on how users interact with public terminals and to find behavioral factors that influence authentication and security. Conducting such work is important since it is the only possibility to observe realistic, unbiased behavior. The differences between the number of users that applied secure behavior during the observations and the much higher amount of users that state secure behavior in interviews and lab studies, impressively highlights this fact.

The findings of the main and the two follow-up studies provided diverse support for criteria presented in previous chapters. The work on VibraPass 3.3.3 showed that relying on "clever" behavior will negatively influence the security of an authentication mechanism. It provided proof that security should not require an active user. The observations of the field study confirmed this and gave further proof. The fact that standard PIN-entry has no whatsoever security built-into it makes it vulnerable to different attacks. Simple measures like hiding the keypad with the non-dominant hand make it much more secure and render most attacks useless. However, only few people apply efficient security measurements when interacting with an ATM. “Security should not require an active user” can therefore be considered an important behavioral criterion when designing authentication mechanisms for public spaces. At the same time, applying this criterion renders learning and teaching approaches unnecessary.

The second criterion that finds strong support in the findings of the field study is about the importance of detailed and precise time measurement as presented in chapter 4. The private nature of the field study did not allow for detailed time measurement of PIN-entry. Still, it showed that there are additional tasks that come with authentication that have to be carefully measured and have to be considered part of it. The need for adequate measurement is further amplified by the fact that authentication is only a minor part of the interaction with the public terminal and thus every measured second counts.

Consistency, as discussed in chapter 4, is an important factor that influences performance and can also influence memorability. In the light of the field study findings, security has to be considered the third important factor connected and indirectly affected by consistency. Due to the fact that memorability issues, even though occurring seldomly, mostly led to security problems within the observations, these findings strongly support the benefits of inner and outer consistency.

The just discussed attributes gave insights on several criteria from a behavioral point of view. In addition to that, two completely new criteria were identified: social compatibility and compatibility to distractions. Both of them are based on behavioral factors that mainly (or solely) occur in public or semi-public settings and are only important due to the presence of other people. Social compatibility takes into consideration that when in the presence of trusted persons, a user might not be willing to implicitly show mistrust to them by applying security measures. The observations and interviews showed that this factor is often responsible for insecure behavior. Designing authentication mechanisms with regard to this criterion can therefore avoid insecure behavior. This criterion and built-in security as proposed by “security should not require an active user” are closely related and solving one can solve both but does not necessarily have to.

Compatibility with distractions refers to findings that diverse distractions, mainly caused by the presence of friends and the like, can negatively influence security as well as performance. Therefore, an important factor of an authentication mechanism is that it has to be easy for a user to recover from errors. At the same time, correctly authenticating cannot demand from the user to constantly concentrate on the authentication process. A countdown-based approach, for instance, can therefore be rejected already in the design stage.

Finally, field studies are difficult to perform and it is hard to derive results from them. During the work performed for these field studies and interviews, we could derive several rules that helped us to improve the study design. For instance, the pre-studies that we performed significantly helped to come up with a study design that allowed for optimized observations. Without such diligent preparatory work, many important results might have been missing. Another lesson learned was that abiding to strict rules was necessary and helpful. Especially dealing with a privacy sensitive context without violating the privacy of the observed users is only possible with a predefined set of strict rules. More and more detailed descriptions of the lessons learned can be found in [33].

Chapter 6

Criteria and Case Studies

> *Es ist nicht genug zu wissen - man muss auch anwenden. Es ist nicht genug zu wollen - man muss auch tun. (Knowing is not enough; we must apply. Willing is not enough; we must do.)*
>
> **– Johann Wolfgang von Goethe –**

Within this thesis, the problem of securing authentication has been approached from diverse angles to understand all factors related to it. Only this way, we will be able to get closer to a solution for this ubiquitous problem: creating a usable and secure authentication mechanism for public spaces with the potential to replace standard authentication.

Within this process, several criteria were identified. Till now, they were only loosely defined and their application was only vaguely hinted. What are the criteria worth if we know them but we do not apply them? As mentioned earlier, the criteria can help to integrate behavioral and technical factors of usable privacy and security into the development process of authentication mechanisms. They can be applied to different phases of the development process, in the design as well as implementation and evaluation phases.

Incorporating security decisions in the development process of IT-systems is not a new idea. However, security is usually only considered from a technical point of view like decisions on data security, encryption etc. The most famous examples for this are most probably UMLsec [62, 71] and SecureUML [85], software engineering methods that use modified versions of the Unified Modeling Language (UML) to model security in the design of a software system. For instance, a software engineer can define encrypted connections when required for the system.

First attempts that try to incorporate behavioral factors to the development process are based on involving users in the design process [1, 49, 50] or use guidelines or recommendations on how to design secure systems that cope with the needs of their users [139]. Due to their nature of being valid for all kinds of security relevant systems, all these concepts stay rather generic and do not provide concrete recommendations but rather hints.

Focusing on an application area, authentication mechanisms for public spaces, enabled us to provide concrete recommendations and criteria rather than generalized models. Therefore, this chapter contributes to this thesis the following way:

a) It lists and summarizes the criteria, their origin and their influence on the authentication mechanism.

b) It describes how and where the criteria can be applied to the development process of an authentication mechanism.

c) A practical example of how to use the criteria based on an authentication mechanism presented in chapter 3 is outlined.

It has to be noted again that this work focuses only on criteria that affect usability, performance and security of authentication mechanisms on a usage level. There are also technical issues of security like encryption, data transmission and the like which are without a doubt very important but which are out of the scope of this work.

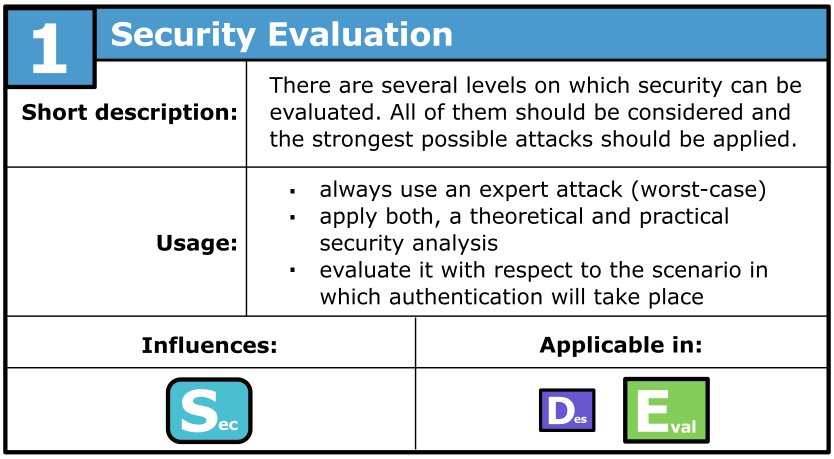
6.1 Criteria

The criteria presented in this chapter are based on results of the diverse user and field studies and on different analyses conducted throughout this thesis. The criteria influence different aspects of authentication mechanisms for public spaces: security, performance and usability. Additionally, they can be applied to the development process in different ways.

For mapping criteria to different phases of the development process of authentication mechanisms, we assume a very generic and coarse process. For this purpose, a very rough categorization is appropriate. We therefore differentiate between three phases: The *design phase* in which the design decisions for the final product or prototype are made. These can be based on paper prototyping, brainstorming or other techniques. The *implementation phase* covers all practical work activities including coding and hardware creation. Finally, the *evaluation phase* consists of different measures to evaluate the functionalities of the system like user studies or beta testing.

6.1.1 Standard Criteria 2.0

The first set of criteria consists of extensions to standard factors that can be found in related work. The extensions refer to deeper insights on the respective factors and recommendations on how they should be extended to gain more valid results on the specific authentication systems.



1	Security Evaluation
Short description:	There are several levels on which security can be evaluated. All of them should be considered and the strongest possible attacks should be applied.
Usage:	▪ always use an expert attack (worst-case) ▪ apply both, a theoretical and practical security analysis ▪ evaluate it with respect to the scenario in which authentication will take place
Influences:	**Applicable in:**
Sec	Des Eval

Figure 6.1: Security evaluation should in any case be conducted by an expert using a worst-case scenario. Both, theoretical and practical security analyses should be conducted. Several decisions in the design phase of the development process are influenced by a theoretical security analysis. The criterion has its biggest effect on the evaluation phase in which all the practical analysis takes place.

1. Security Evaluation

As seen during the evaluation of related work in chapter 2, there is no standard on how the security of an authentication mechanism is evaluated and analyzed. This is critical for two reasons: It makes comparing different systems to each other difficult and makes it hard to judge the real security of an authentication mechanism. The studies performed during this thesis gave diverse insights on how this can be improved and provided us with the possibility to give strict recommendations on security evaluation.

Firstly, there is the question of using theoretical or practical security analyses. The evaluation of the different authentication mechanisms presented in chapter 3 highlighted that both are necessary. While a theoretical security analysis can be applied very early in the development process, already during the design phase, practically analyzing security takes place in the evaluation phase. Applying both approaches thoroughly can reveal very interesting facts, especially in combination. It is possible that a theoretical high security does not hold against real attacks. For instance, the extremely large password space of ColorPIN (chapter 3.4.1) turned out to become a security problem in the practical security evaluation since it made the system highly vulnerable to intersection attacks.

The second question that comes to ones mind when reading related work is whether a practical security analysis should be performed by study participants or by an expert user. Again, the work on authentication mechanisms conducted within this thesis confirms that only an expert attack by a person familiar with the system can deliver appropriate results and enable worst-case attack scenarios like camera surveillance in case of many systems.

Finally, the security of an authentication mechanism should be evaluated in the scenario for which it has been designed. For instance, in a public space as defined in chapter 1.1, the risk of shoulder surfing is rather high and should thus be part of the evaluation. Another example to the respective scenario are skimming attacks, that is manipulations of the terminal.

The recommendations on security evaluation are summarized in figure 6.1. Theoretical and practical security analyses should be applied and if available performed by an expert rather than user study participants (for instance done in [74]). This approach proved beneficial in the evaluations of the authentication mechanisms presented in chapter 3. Additionally, a worst-case scenario should be chosen with respect to the scenario. Security evaluation mainly takes place in the evaluation stage of the design process. Most parts of a theoretical security analysis, however, can be performed in an early design stage. For instance, the decision on which authentication token to choose is highly influenced by such an analysis.

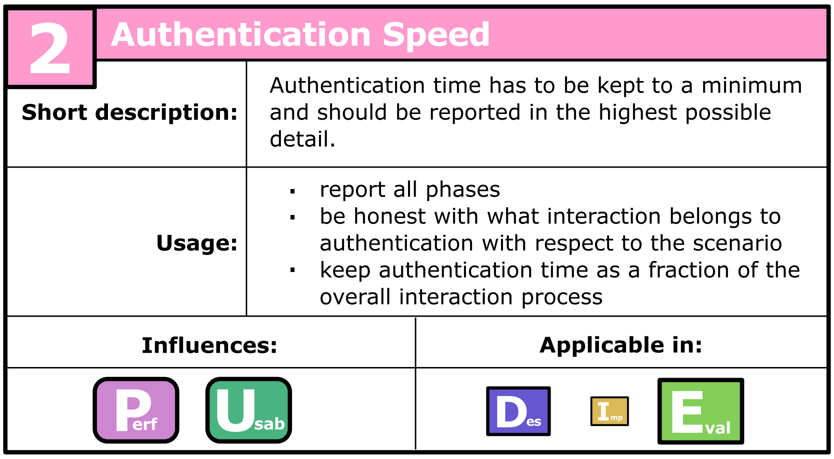
2. Authentication Speed

Authentication speed is the number one factor in current evaluations of authentication mechanisms. It is very uncommon to find related work that does not report speed. All researchers seem to agree on the importance of it, but there is no common sense about what times are acceptable. Furthermore, speed is reported very differently and, just like security, makes it very difficult to compare mechanisms. Therefore, we redefined the approach on evaluating and reporting authentication speed.

In the chapter on authentication mechanisms (chapter 3) and the long-term and field studies (chapter 4 and 5), we could gain several insights on the importance of correctly measuring and reporting authentication speed. In chapter 4.2.2, we proposed a speed measurement approached based on different phases. We argue that such an approach should always be taken. Besides the nice side effect that it makes authentication mechanisms much more easily comparable, it can help to reveal highly important insights, especially considering learning effects as shown in chapter 4.

Related to this is the need to be honest about what parts of the interaction are part of authentication and thus have to be reported as part of its phases. For instance, the work on MobilePIN (chapter 3.3.2) showed that connection establishment is a very critical aspect and the biggest disadvantage of authentication mechanisms based on user owned devices. Depending on the usage scenario, this time has to be counted as part of the authentication itself. If the interaction itself requires a connected mobile device, this time can be considered as part of the main interaction and not as an overhead to authentication.

The hardest question to answer is on the right authentication speed. We argue that it is not possible to give a fixed value in terms of seconds since it is heavily depending on the users and the context of the interaction. However, all the work of this thesis indicates that authentication speed has to be kept very high so that the authentication process only takes a minor fraction of the overall interaction process. Violating this will lead to the users rejecting the system.



2 Authentication Speed

Short description:	Authentication time has to be kept to a minimum and should be reported in the highest possible detail.
Usage:	▪ report all phases ▪ be honest with what interaction belongs to authentication with respect to the scenario ▪ keep authentication time as a fraction of the overall interaction process
Influences:	**Applicable in:**
Perf Usab	Des Imp Eval

Figure 6.2: Criterion two influences performance and usability of an authentication mechanism. It is important to report all phases of authentication and be honest with what steps of the interaction are part of authentication process. This criterion can be applied during design, implementation but mostly during evaluation.

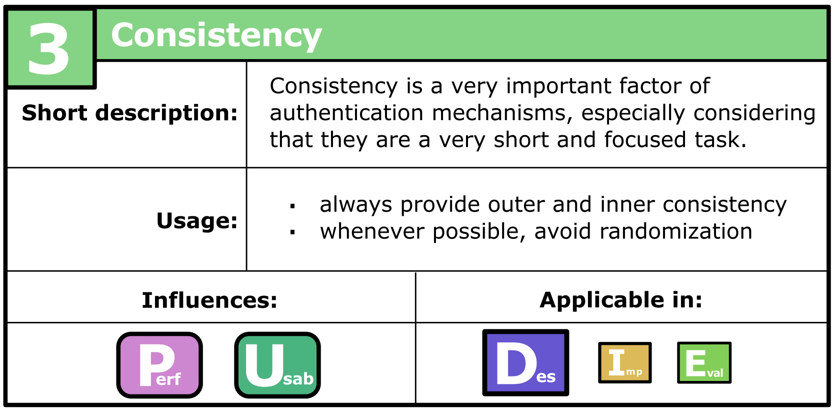
As summarized in figure 6.2, it is important to report all phases of authentication and be honest with which interaction is part of the authentication process. How well this criterion is implemented influences both performance and usability of the system. Realizing this criterion, involves all parts of the development process. Most effects, especially long-term and learning, can only be analyzed during the evaluation phase.

3. Consistency

In contrast to standard user interface design, consistency was never considered a very important factor for authentication mechanisms. In chapter 4, we could show that inner as well as outer consistency are of high importance with respect to performance. As mentioned earlier, outer consistency refers to the consistent use of the same authentication interface or mechanism (e.g. the same key arrangement). Even though it should be implicitly assumed, it is often neglected. The fact that in real world settings of ATMs different kinds of keypad layouts can be found, highlights that this is often not dealt with appropriately.

Furthermore, lack of consistency has negative effects on memorability which can also lead to security issues, for instance, if users write down their authentication tokens (e.g. their PIN) since they cannot remember them. Therefore, consistency should be an essential part of an authentication mechanism.

Stating that consistency is an important factor of an authentication mechanism additionally means that if possible, randomization should be avoided. This can be problematic since randomization is an often proposed tool to improve security. However, the work on ColorPIN (see chapter 3.4.1) showed that the negative effects of randomization can be reduced with clever design by providing partial inner consistency.



3 Consistency

Short description:	Consistency is a very important factor of authentication mechanisms, especially considering that they are a very short and focused task.
Usage:	▪ always provide outer and inner consistency ▪ whenever possible, avoid randomization

Influences:	**Applicable in:**
Perf Usab	Des Imp Eval

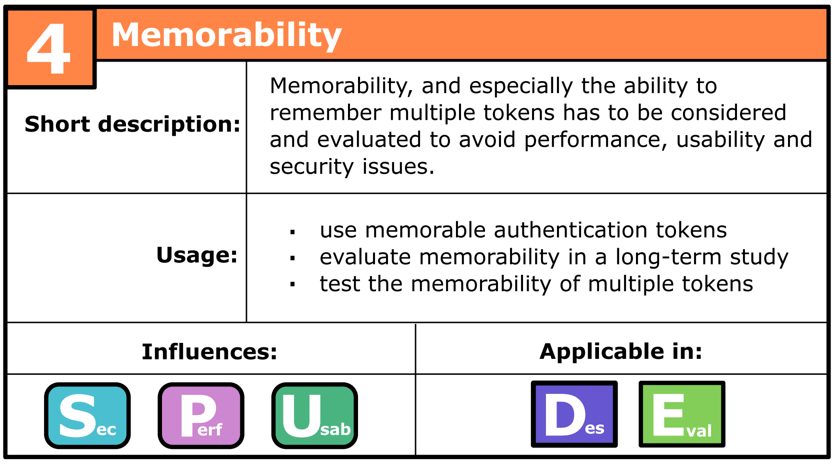
Figure 6.3: Consistency influences performance and usability of an authentication mechanism. The two main rules are to provide outer as well as inner consistency and to avoid randomization wherever possible. Consistency is mostly important to consider during the design phase but can also be influenced during the implementation and its influence should be evaluated.

The recommendations are outlined in figure 6.3. The most important rules of this criterion are to provide outer as well as inner consistency. Furthermore, this means to avoid randomization wherever possible. Sticking to this criterion will not only improve usability but also the performance of the authentication mechanism. Consistency is mostly defined and realized during the design phase but can be influenced during the implementation phase as well. If consistency cannot be fully applied to, the influence of its lack should be carefully evaluated.

4. Memorability

Memorability is the last criterion that represents an extension of a commonly applied criterion. An important finding is that it is influenced by many factors like consistency. As shown in chapter 4.4.2, lack of consistency limits the use of advanced learning strategies and therefore has a negative influence on memorability. Besides these factors, the choice of the authentication token (the authentication credential) can be considered as the factor that influences memorability the most.

Even though it is of great importance, memorability is seldom evaluated in literature on authentication mechanisms. The main reason for this is that such an evaluation can only be done using a long-term user study. Lack of time and resources often make this difficult. This becomes even worse when considering that an optimal evaluation of memorability should include the use of several different instances of authentication tokens like PINs or passwords. This way, it reflects the condition as we find it in real world settings were users have many different accounts with different tokens. Therefore, memorability is often only approached on a theoretical level which cannot cope with memorability influencing factors like consistency. Therefore, whenever possible, long-term, multiple token evaluations of memorability should be conducted.



4 Memorability

Short description:	Memorability, and especially the ability to remember multiple tokens has to be considered and evaluated to avoid performance, usability and security issues.
Usage:	▪ use memorable authentication tokens ▪ evaluate memorability in a long-term study ▪ test the memorability of multiple tokens

Influences:	Applicable in:
Sec Perf Usab	Des Eval

Figure 6.4: Memorability influences security, performance and usability of an authentication mechanism. It is mostly affected by design choices and should be evaluated using multiple tokens if possible.

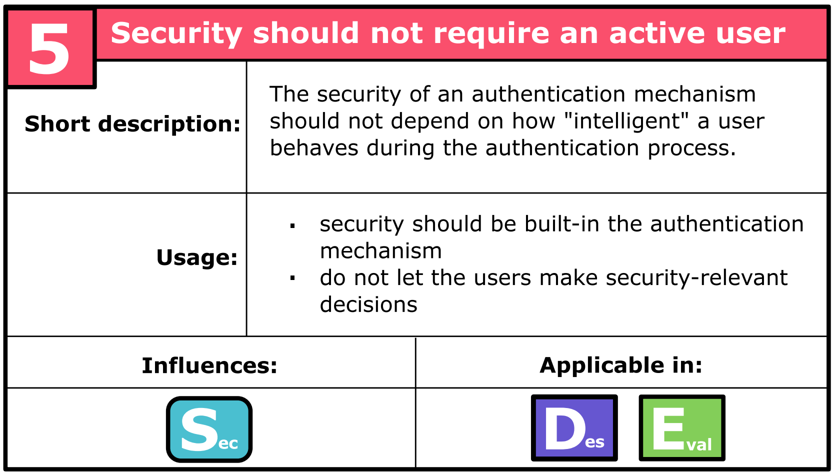
As seen in the field study on ATM use in chapter 5, memorability problems can cause insecure behavior on the user side like writing PINs down. Such observations highlight the importance of the appropriate choice of authentication token. This is something that has to be decided upon early in the design process.

Criterion four is summarized in figure 6.4. Memorable authentication tokens should be chosen and evaluated in long-term studies with several authentication tokens. The decisions made in the design process influence security, performance and usability of the authentication mechanism.

6.1.2 New Criteria

The previously presented criteria can be seen as improvements or extensions to (more or less) standard criteria. That is, they are considered widely in research on authentication mechanisms. Nevertheless, they are usually only applied in a vague way. The proposed improvements (one could also say stricter versions) are helping researchers to gather further insights on their own work and avoid design mistakes already in very early stages of the development process.

In addition to those, we found several criteria that are hard or impossible to find in related work and that are definitely not part of the standard development processes of authentication mechanisms. These remaining criteria describe behavioral factors in authentication and are partially based on the presence of other persons. Therefore, they are especially important for authentication in public spaces. It would have been nearly impossible to get the necessary insights to define them without conducting the long-term field studies and interviews presented in chapter 5.



5 Security should not require an active user

Short description: The security of an authentication mechanism should not depend on how "intelligent" a user behaves during the authentication process.

Usage:

- security should be built-in the authentication mechanism
- do not let the users make security-relevant decisions

Influences:	Applicable in:
Sec	Des Eval

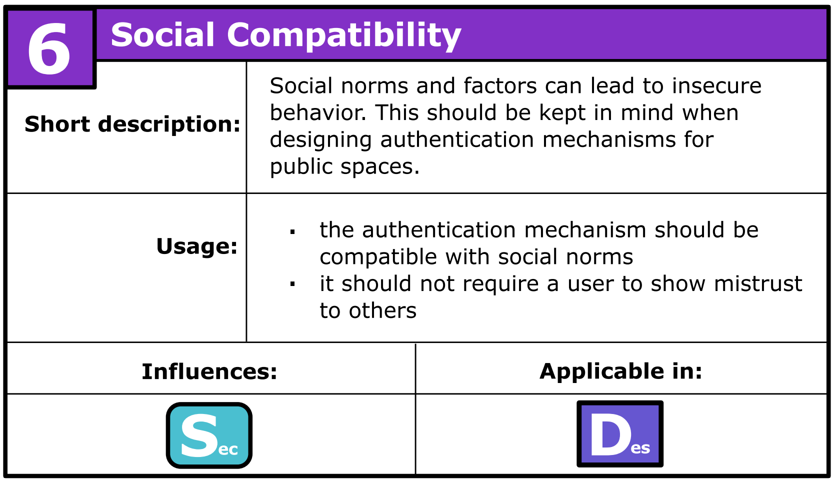
Figure 6.5: Security should not require an active user. This means that security should be built-in to the authentication mechanism and security relevant decisions should not be made by the users. This criterion influences the security of the authentication mechanism. It is mostly affected by design choices and should be evaluated thoroughly in the evaluation phase.

5. Security should not require an active user

The first criterion deals with the users as a possible security risk based on their skills and willingness to behave securely. The first indication that security should not rely on the users' skills was found during the work on VibraPass (chapter 3.3.3), or better said the observations on “bad lies” as discussed in chapter 3.3.3. It turned out that too much responsibility for the security was given to the users which resulted in insecure behavior that an attacker could exploit to identify the users' PINs. Better said, the quality of “lies” that the users employed decided upon the security of the system. That is, an authentication system should not require from the user to actively support security.

This assumption is strengthened by the observations made during the field study on ATM use. Firstly, it showed that the majority of users does not apply security measures, like hiding their PIN, when using an ATM. That means that the extra effort of making the interaction (more) secure is simply ignored. Furthermore, physical constraints can make secure behavior more difficult or impossible and thus, users will not apply it since security is seldom a user's primary goal [65, 133].

The recommendations on this criterion, as shown in figure 6.5, are therefore as simple as effective. Security decisions should not be made by the users of an authentication mechanism. That means that security should be a built-in property of the authentication mechanism and should not depend on clever behavior of the user. This has to be handled carefully, however, since it can come at the cost of usability. This criterion has a large influence on the security of a system and should already be considered during the design stage and should be evaluated carefully.



6 Social Compatibility

Short description:	Social norms and factors can lead to insecure behavior. This should be kept in mind when designing authentication mechanisms for public spaces.
Usage:	▪ the authentication mechanism should be compatible with social norms ▪ it should not require a user to show mistrust to others

Influences:	**Applicable in:**
Sec	Des

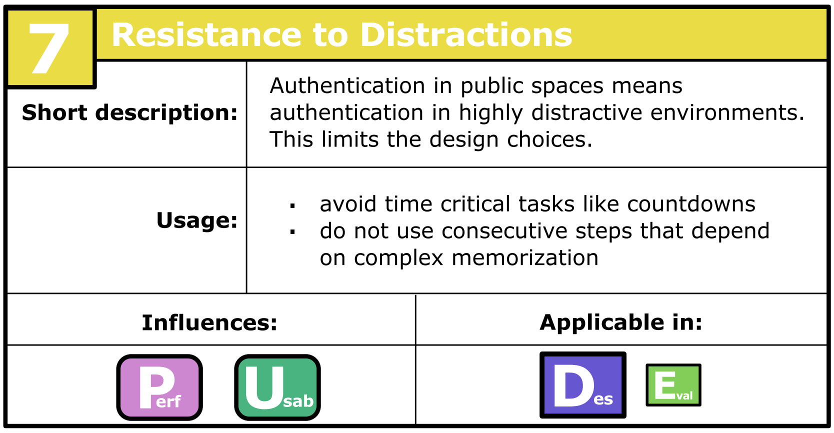
Figure 6.6: Social Compatibility means that the authentication mechanism should should not require the users to show mistrust when securing the input. This criterion highly influences the security of the authentication mechanism and should be considered during the design phase of the development process.

6. Social Compatibility

The public nature of authentication in public spaces leads to its very specific problems requiring very specific criteria. Compatibility with social norms and factors is one of these criteria that is of high importance whenever a user is in company with other people and has to perform authentication. Even though there are established social norms related to authentication, like looking away to give the user the necessary privacy, trust or the lack of it can have a much higher impact. That is, in many instances, users do not want to show mistrust to people around them and therefore apply insecure behavior.

This problem can partially be solved by fulfilling criterion 5 (security should not require an active user). However, to fully cope with this problem, an authentication mechanism has to be designed to be compatible with such social factors and norms. That is, it should not require the user to hide the input from others which could be interpreted by them as a sign of mistrust. If, for instance, hiding is part of the authentication process without which it will not work, this action cannot be considered as a proof of mistrust. Such a system would therefore fulfill criterion 6.

As summarized in figure 6.6, the basic principles of this criterion are that an authentication mechanism should not require a user to show mistrust to others and therefore should be compatible with social norms. The observations of the ATM field study, as presented in chapter 5, highlight that violation of this criterion highly influences the security of the authentication mechanism. Therefore, extra care of this aspect has to be taken during the design stage.



7	Resistance to Distractions
Short description:	Authentication in public spaces means authentication in highly distractive environments. This limits the design choices.
Usage:	▪ avoid time critical tasks like countdowns ▪ do not use consecutive steps that depend on complex memorization
Influences:	**Applicable in:**
Perf Usab	Des Eval

Figure 6.7: Resistance to distractions means that the authentication mechanism should work even in highly distractive environments such as public spaces. For instance, time critical tasks with a time limit should be avoided. This can be applied during design and should be evaluated. Its application influences performance and usability.

7. Resistance to Distractions

The last criterion is based on the same ground truth as criterion 6. The fact that the authentication takes place in a public setting makes it vulnerable to all kinds of distractions. This can be people talking to the user, noise, traffic and other kinds of distractions. The conclusion of this is that authentication cannot require the undivided attention of a user and should enable the user to easily recover from distractions. Standard PIN-entry fulfills this requirement since it is hardly difficult to input the next consecutive digit when interrupted for a certain amount of time. In the worst case, the authentication attempt can be restarted.

The implications are described in figure 6.7. To make an authentication mechanism resistant to distractions it should avoid the use of consecutive steps that depend on each other and might be hard to be remembered if not executed directly when required. Additionally, time critical tasks, for instance depending on countdowns, should not be used. Sticking to these rules positively influences performance and usability of the system. The phase in which it can be mostly applied is the design stage. In an evaluation, the success of the designed approach can be analyzed by confronting users with distractions or using a potentially distractive environment.

6.1.3 Criteria in the Development Process

The description of the criteria already contained information on where to apply the criteria within the development process of an authentication mechanism. The reason why few criteria can be applied during the implementation phase lies in the fact that they are mainly basic design choices that can neither be positively nor negatively influenced during implementation. Other factors focus on the evaluation process and enable researchers to gain advanced insights on security,

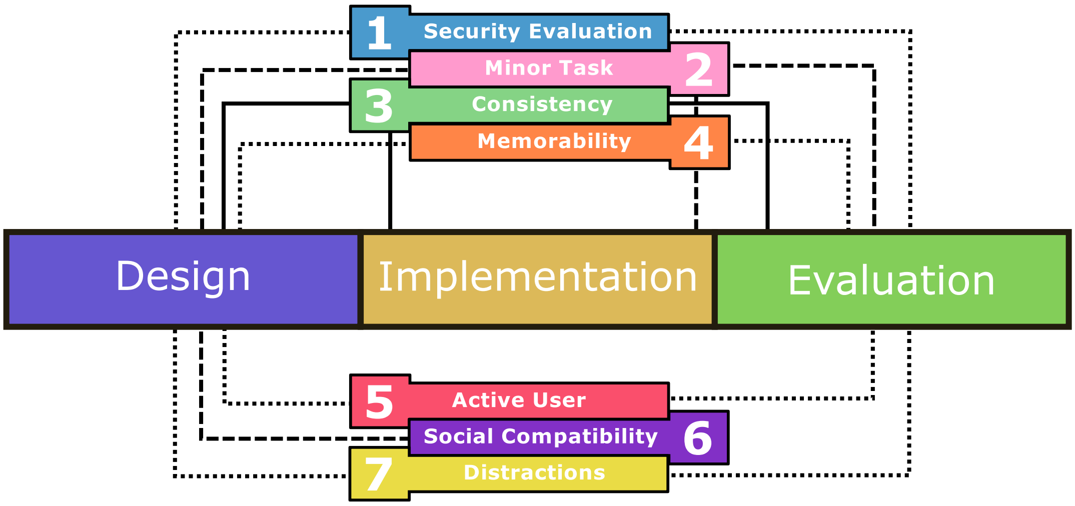


Figure 6.8: The different criteria in the development process. Design and evaluation are the phases which are influenced by the different criteria the most. Implementation is only partially affected.

performance and usability of their systems. The implementation phase only plays a major role when it comes to evaluation-sensitive criteria. For instance, the phase-based time measurement proposed in this thesis requires a well thought of implementation and special considerations to correctly record all times. Therefore, this criterion also influences implementation.

Figure 6.8 provides a quick overview in which part of the development process the different criteria should be applied. This schematic model does not consider possible cycles in the development process as known from the waterfall or other software development models. There are several correlations between the different criteria. Adhering to criterion 5 can partially solve and influence social compatibility. Another example is the relationship between consistency and memorability. However, these correlations are not considered in the diagram.

6.2 Case Studies

Having the detailed description of the criteria and knowing to which parts of the development process they can be applied to, it seems necessary to further describe their actual use. Therefore, this chapter will outline examples of how to apply the criteria to the creation process of authentication mechanisms. We will exemplarily describe the usage based on examples of authentication mechanisms created during this work (see chapter 3).

These case studies will not only highlight the benefits of the criteria, extended and new, but also provide proof for the importance of applying them in different phases of the development process. For instance, it would have been hard to find out about “bad lies” during the design process of VibraPass. Therefore, this criterion requires not only good design but also thorough evaluation.

One problem remains: Large parts of the findings that the criteria are based on were found during the development of the authentication mechanisms. Therefore, the criteria did not exist when the authentication mechanisms were created. This means that parts of the process can only be "simulated" within this chapter. Simulating refers to the theoretical application of the criteria and how it would have influenced the final prototypes. Additionally, not all the logging data allows for analyzing the data as proposed in our phase-based time measurement approach. Summarized that means that in some instances, only potential benefits of the criteria can be discussed. This "what-if" scenario will therefore also reflect on how the criteria would have influenced the work on the different authentication mechanisms.

6.2.1 Case Study 1: VibraPass

VibraPass, as presented in chapter 3.3.3, is an example for an authentication mechanism based on hardware owned by the user. The evaluation was based on a user study and showed good performance with high security. However, several weaknesses were revealed as well. The security of VibraPass strongly depends on the users' behavior and the system lacks inner consistency which negatively influences performance and usability. Additionally, the system requires a connected device which might result in a major overhead.

The question to be answered in this chapter is whether some or all of the drawbacks of the system could have been avoided by applying the criteria. Maybe the concept should have even been rejected in an early design stage.

Design Phase

The security analysis of the design phase was conducted with respect to the methods recommended in criterion 1 and attested extremely high security. VibraPass theoretically is resistant to skimming and even man-in-the-middle attacks. The only main weakness the theoretical analysis revealed are intersection attacks. Therefore, the system would have passed the design stage with respect to security (criterion 1).

One important aspect that was missing in the design phase of VibraPass was a realistic evaluation of authentication speed. In the original design process, only the marginal overhead caused by the "lie overhead" was considered. The additional major overhead caused by connection establishment time was simply neglected. Within this thesis, we saw that this is important to consider and should have been part of the design phase. This would have revealed that VibraPass requires a connection of some kind. However, considering the fact that no sensitive data is transmitted over the wireless channel, an insecure and thus more efficient connection mechanism (e.g. based on NFC) could have been used. Even though being much faster, also this interaction would cause an overhead of several seconds and should have been counted as part of authentication, depending on the scenario. Since VibraPass was originally designed for ATMs, the connection is not part of the actual interaction and only part of authentication and thus has to be counted part of the preparation phase (see chapter 4.2.2). With respect to criterion 2, VibraPass should have been more critically assessed in the design phase which could have led to some redesign.

For criterion 3, consistency, the result of the design phase would have been quite definite. The concept clearly contradicts the recommendation that randomization should be avoided. Therefore, this concept might have been a candidate for rejection. However, the randomization does not cover the whole interaction but only a fraction of it. That is, would this criterion have been applied, the result could have been to take care about this factor during evaluation to figure out if the remaining consistency is enough to legitimate the appropriateness of VibraPass.

With respect to memorability, in the original design phase it was assumed that it should be similar to standard PIN-entry besides the lack of consistency which was not assumed a major issue at that time. Now, we know that this lack is very likely to have major influence on memorability of the authentication token PIN, mainly because it eliminates the use of advanced memory strategies. This means that also for criterion 4, the system would have gotten bad grades.

The same accounts for criterion 5, for which the system has never been evaluated against. The grade of security depends on clever behavior of the user. That is, security is actively influenced by how the user behaves. Therefore it violates criterion 5. Nevertheless, even a thorough analysis with this criterion in mind would not have been 100% likely to reveal this factor. It can be assumed that similar to the original approach, this could have only be revealed using a thorough analysis of the user study data.

Finally, by forcing the users to lie, the system is clearly compatible to social factors and therefore in line with criterion 6. The last criterion, resistance to distractions, is not fully fulfilled. VibraPass requires the user to stay attentive during the whole authentication process and does only partially allow for interruptions. This is mainly due to the fact that the input is different every time since lies are randomly placed within the PIN. Without knowing this criterion, this fact could have only be revealed using a long-term field study in which the system is actually being employed by the users on public terminals.

The quality of the original results of VibraPass can be partially attributed to the lack of a long-term study which could have revealed memorability and performance issues. The analysis based on the seven criteria showed that it would have been a candidate for rejection during the design stage. However, the analysis of the design stage also revealed major benefits of the system security- and performance-wise, especially compared to related work. Therefore, instead of rejection, the favorable approach would have been to take these factors into account when evaluating the system. Additionally, some of the criteria could have been fixed during the design stage. For instance, to fulfill criterion 7 (resistance to distractions), a history functionality could have been added to the prototype.

Implementation Phase

The implementation phase of VibraPass would have experienced one major change with respect to criterion 2. To appropriately measure all the different phases, the logging of times would have required some extension. In the original code, only active authentication was recorded. There is no functionality in the prototype to adequately measure preparation and confirmation. Also consistency (criterion 3) was not considered. However, in the case of VibraPass, the implementation could not have improved or worsened it significantly.

Evaluation Phase

Security of VibraPass was conducted strictly adhering to criterion 1 (even though it did not exist at that time). Firstly, we used a worst-case attack with respect to the scenario of a public terminal. The attack included multiple video as well as sound recordings. Additionally, the analysis of the material was performed by an expert who was part of the group that worked on VibraPass. We argue that only this way, “bad lies” could be identified as a security issue.

The picture is different when considering criterion 2. The implementation of VibraPass only allows for detailed insights on the active authentication phase. This is also related to the study setup which did not allow for recording preparation and cleanup times. It can be assumed that preparation is rather short since due to randomization there is not a lot the user can mentally prepare to. Still, this aspect would have deserved attention and might have led to important findings.

In the analysis, the influence of lacking consistency was not adequately reflected. A deeper look at the data reveals an interesting finding. For all cases with a lie overhead greater than 0%, the per-digit speed decreases drastically. While in the standard PIN-entry condition, the input of one digit takes 0.56 seconds on average, this time increases to 0.78 seconds in the 30% overhead condition. This is an increase of around 39% per digit and a good indicator that the performance is negatively influenced by the lack of inner consistency. That is, applying criterion 3, as done in this chapter, would have revealed a huge relative decrease of authentication speed. This is an interesting finding even though the absolute authentication speed-to-security ratio of VibraPass can still be rated very good. That is, the loss of speed that comes with applying this technique is minimal compared to the gained security.

Originally, we assumed that memorability would be similar to what we know from standard PIN-entry. Now, with the results from the long-term study on PIN-entry, we know that lack of consistency can influence this factor as well as performance. Therefore, VibraPass should have been evaluated using a long-term study, optimally using multiple tokens. This could have revealed major memorability flaws. It is also possible that the lack of consistency and the specific setup of VibraPass could have led to different memory strategies.

Without knowingly applying it, in the original evaluation of VibraPass, criterion 5 has been thoroughly evaluated. This was done by performing an in-depth analysis of reasons for why the system could be broken in some instances, that is, why security failed. It turned out that too much of the responsibility of its security lies in the hands of the users. That is, the evaluation revealed security flaws directly caused by this fact.

Finally, criterion 7, resistance to distraction, was not evaluated. Knowing this criterion, an evaluation would have been straight forward. As part of the user study, or ideally during a follow-up study, the setup should have involved a task where the authentication has to be stopped and continued from the previous point. The original study already provided indications that this is rather difficult.

Interestingly, some of the criteria were already implicitly applied during the original evaluation. On the other hand, just like during the design phase, some important criteria were missing. Even though the data was partially available (e.g. data on the influence of missing consistency), it was originally not analyzed. Even applying the criteria afterwards, in this case study, shows how important they can be and what further insights they can generate.

Conclusion

The results of this case study show that with the help of the criteria, several of the issues that were discovered during the evaluation of VibraPass could have been identified already in an early design phase and therefore effort could have been saved. Additionally, the application of the criteria reveals open issues like the lack of an evaluation of resistance to distractions.

The remaining question is how to handle situations in which some of the criteria are violated. Should VibraPass have been rejected during design and never been implemented? There are several indications that rejection would have been a mistake. Firstly, even though some of the criteria were partially or completely violated during the design phase, performance and security were rated highly positive. Most importantly, however, instead of rejecting the concept, it could have been improved easily with respect to the criteria. The basic concept of VibraPass allows the conclusion that it is not resistant to distractions. As for any iterative design process, this insight could have been used to add functionality to make it fulfill this criterion, for instance by adding some kind of lie history to the mobile device or more securely by repeating the vibration after a specific timeout. Using the second approach, users would know that if the vibration repeats, the next turn is a lie and if it does not repeat within say three seconds, the current digit is not supposed to be a lie. This means that the criteria can also be counted as valid tools that support iterative software development approaches.

6.2.2 Case Study 2: EyePassShapes

In chapter 3.2.4, EyePassShapes was presented, an authentication mechanism based on PassShapes [36, 132] and gaze gestures [43]. Within the categorization of authentication mechanisms for public spaces, it falls into the category of hardware based approaches, using an eye tracking device mounted to the public terminal. The system was attested very high security and good performance with a small trade-off between those two factors. That is, practical higher security was achieved by using the slower input variant of several consecutive strokes rather than using a single-stroke and thus faster attempt.

We argue that within this thesis, EyePassShapes was one or maybe the best candidate with a realistic potential to replace standard PIN-entry for public terminals due to very good results in performance, usability as well as security. It has to be noted that the accessibility of the system is limited to people with normal seeing ability and motoric skills.

As for VibraPass, the criteria did not exist in their current form when designing, implementing and evaluating EyePassShapes. Therefore, they were not explicitly considered. However, there were already many lessons learned from earlier designs. In this case study, we will take special care on analyzing whether the good results of the system cope with the application of the criteria. That is, whether the system implicitly fulfilled the criteria and whether the good results can be attributed to this fact.

Design Phase

The theoretical security evaluation of the original design phase of EyePassShapes was carried out as proposed in criterion 1. A thorough theoretical security analysis was conducted and high security was attested to the system. The main weakness of EyePassShapes is advanced skimming attacks based on multiple cameras that are very unlikely in a public setting.

The influence of criterion 2 on the design process is defined by honestly judging the possible performance in sense of authentication speed. In contrast to VibraPass, in the design phase of EyePassShapes, this aspect has been fully adhered to. In fact, one of the advantages of the concept was the lack of a calibration process. This makes it faster since such a process would have to be counted as part of the authentication attempt if eye tracking itself is not required for the interaction with the public terminal. Using PassShapes as very fast authentication tokens supported this approach. The only open question was whether the same performance and usability advantages apply for gaze interaction as well.

EyePassShapes provides strict outer and inner consistency. Even more than that, it uses an authentication token that has been designed to provide advanced memorability by using consistent stroke based shapes that have to be drawn in exactly the same order of strokes every time the user authenticates with the system. Randomization has been completely avoided as well. Therefore, again implicitly, the decisions made during the design phase fulfill the recommendations of criterion 3 and 4 since with PassShapes, an authentication token with proven positive memorability qualities has been chosen.

Criterion 5 states that security should not require an active user. Using gaze-based input makes the system resistant to manifold attacks which cannot be negatively influenced by the user by not using gaze input. However, during the evaluation of the system, a security advantage of EyePassShapes executed in several consecutive strokes could be attested. Thus, the level of security can be influenced by the users' behavior. As opposed to VibraPass, it is not possible to use EyePassShapes in a way that simply gives away the authentication token, meaning that the security of EyePassShapes cannot be "turned off". From a design phase point of view, this criterion has to be considered fulfilled.

For social compatibility, the situation is much simpler. To authenticate using EyePassShapes, the users have to use their eyes, secretly transmitting the information to the terminal. That is, any friends or other people near the user cannot see the authentication but at the same cannot blame the user for this fact since there is no alternative to this behavior. This criterion is thus the next that has been implicitly applied during design process. This is also an example of how solving criterion 5 can implicitly solve criterion 6 as well.

Finally, the last criterion that is of importance during the design process is resistance to distractions. That mainly means that it has to be possible to easily continue authentication once it has been interrupted due to distractions. One of the design decisions made for EyePassShapes was that the single strokes of the PassShape can be executed in arbitrary consecutive chunks. Even though the possibility is low that a user will be interrupted within the five seconds that it takes to authenticate in a single stroke, it is theoretically possible to stop authentication and continue from where it stopped. Based on the fact that the shape is always the same and no randomization is used, this is feasible to achieve by a user. That is, also the last criterion has been integrated into the design of EyePassShapes.

As mentioned before, performance and security of EyePassShapes are very promising. Interestingly, all the criteria of the design process were implicitly considered in its concept. We argue that this involuntary fact is responsible for the positive attributes of the system. This also shows that much of the positive and negative properties of an authentication mechanism are already decided during the design stage.

Implementation Phase

For measuring authentication speed, only the authentication phase and confirmation were recorded. Using gaze-based technology, it would have been rather easy to record preparation. As opposed to most systems, no “tricks” would have been required but simple observing the gaze before the actual interaction. Consistency is already fully considered by the design and the implementation could only minimally influence it. A technical evaluation of different implementation alternatives has been conducted for this purpose. Based on this, a background picture was selected, providing additional consistency.

Evaluation Phase

In all work on authentication mechanisms presented in this thesis, an appropriate and in-depth security evaluation was always one of the most important factors. Therefore, as demanded by criterion 1, thorough theoretical as well as practical security analyses and evaluations were conducted. The attacker was an expert on EyePassShapes who employed a worst-case scenario based on video recordings with the information on when the authentication began and when it ended. This way, the difficult relationship between security and authentication speed could be revealed. Thus, like for the design phase, criterion 1 was fulfilled for the evaluation phase as well.

Since recording the preparation phase was neither considered during design nor implementation, it was not recorded during the evaluation as well. Besides this, all times were precisely recorded and compared. To record preparation, the time from the moment the background picture was shown to the first button press could have been used. We assume that the time is similar to the preparation phase of standard PIN-entry as presented in chapter 4. It often consists of a mental task in which authentication is performed or the PIN is recalled using a memory strategy. Without having this data, we cannot say this for sure and thus, criterion 2 was only partially implemented in the evaluation.

The EyePassShapes prototype provided full outer and inner consistency (criterion 3). This means that no negative effects due to its absence could be measured. Consistency played an important role of the long-term memorability evaluation of the system and we argue that it is one of the reasons for the good memorability attributes of EyePassShapes.

Criterion 4 states that an authentication mechanism should use a memorable authentication token, for instance, a token that exploits the user's muscle memory. This was already considered during the design phase and was the main reason why PassShapes were chosen for the system. Additionally, it requires long-term evaluations of the system, in the best case with multiple authentication tokens. Within the evaluation of EyePassShapes, this was only partially fulfilled by applying a long-term study without using multiple tokens. In this study, the system was attested very good memorability properties.

As mentioned in the analysis of the design phase, the criterion that security should not require an active user was fulfilled. The evaluation of the system showed that security cannot be turned off by insecure behavior during the interaction. Nevertheless, an in-depth analysis of the data of the security evaluation revealed that the degree of security can be influenced by one basic decision: performing the authentication in one or in several consecutive strokes. While the latter approach was more secure, one-stroke interaction was faster. In both cases, the system was highly secure. This is a major difference to VibraPass for which the evaluation showed that security can basically be completely turned off by insecure behavior. Again, this criterion can be considered fulfilled for EyePassShapes.

The last criterion that has an influence on the evaluation of EyePassShapes is resistance to distractions. As for VibraPass, this has not been considered during the evaluation. There was no extra condition in which this resistance was tested. Theoretically, a high resistance can be assumed but it has not been proven and thus this criterion has not been considered in the evaluation phase. In [44], the authors describe a possible solution for this problem. They used a system of surrounding screens to simulate a realistic ATM setting. Inserting explicit distractions in such a setup could be a good candidate to fulfill criterion 7.

Summarized, only criterion 7 was not included in the evaluation process of EyePassShapes at all. In addition to that, criterion 2 was only partially evaluated since times for preparation were not recorded.

Conclusion

The most interesting outcome of this case study was on the very good performance of the system even though the criteria did not exist when we conducted the work on EyePassShapes. We could show that during the whole development process, most criteria were implicitly applied. We cannot attribute this to the specific development process we used since it did not noticeably differ from the one that was applied to VibraPass. EyePassShapes, as a concept, proved to be very

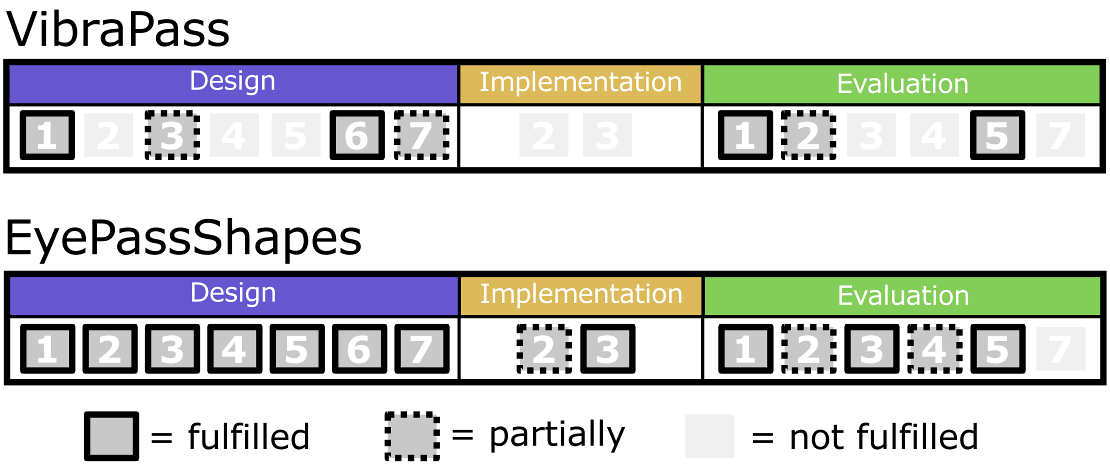


Figure 6.9: The criteria applied to the different development stages of VibraPass and EyePassShapes. The comparison shows that VibraPass only fulfills part of the criteria while EyePassShapes fulfills most of them. This result correlates with the overall ratings of the systems based on their original evaluation.

efficient and a promising candidate for an authentication mechanism for public spaces. That is, if the criteria would have existed at that point, the design phase would have already revealed its positive properties and would have given first indications that we were “on the right way”.

It was not a lucky accident that the criteria were applied but rather an effect of the good concept. Furthermore, several lessons learned from previous work positively influenced the design of EyePassShapes. We argue that this approach can work the other way round as well. If EyePassShapes is good since the criteria are passively fulfilled, an authentication mechanism can be improved and appropriately judged by actively assigning the criteria as well. This can be partially seen in case study 1, VibraPass (chapter 6.2.1), in which improvements to the concept based on the criteria were proposed.

6.3 Lessons Learned

In this chapter, we presented and described seven criteria for developing authentication mechanisms for public spaces. Criteria one to four present extensions of standard criteria and redefine usability issues and measurement. In addition, criteria 5 to 7 are new within the development process and highlight social and behavioral properties. The criteria are based on the development of authentication mechanisms and extended by several field and long-term studies.

To get a better understanding of how the criteria support the development process and how they influence the quality of an authentication mechanism, we retrospectively applied them to two examples of authentication mechanisms presented in chapter 3. The first one, VibraPass, possesses some major flaws like the possibility to perform “bad lies”. EyePassShapes, on the other hand, could be identified as a very good candidate. These findings from the original evaluations and especially the overall quality of the systems highly correlate with findings from the case studies about the degree in which the system fulfills the seven criteria in the different development stages. An overview is depicted in figure 6.9. The view on this schematic overview reveals this fact.

Our experience throughout this whole work as well as the results from the case study analysis give strong support to what we claimed at the beginning of this thesis. The criteria provide useful help in the design process of authentication mechanisms. They help to judge concepts in a very early design stage and more importantly help to improve them to avoid design flaws that can lead to decreased security and usability. The criteria furthermore influence the evaluation of authentication mechanisms for public spaces and, once applied, make them more easily comparable (see again figure 6.9). That is, the criteria allow for a more critical view on the concept and performance of a system.

When evaluating an authentication mechanism with respect to a specific scenario, all criteria have validity and have to be applied. However, the criteria have to be differently assessed in different settings. As mentioned before, in a scenario, which requires a connected mobile device, connection times can be counted as part of the primary interaction, while in other scenarios, it is part of the authentication process. Another example is the evaluation of distractions. While in an ATM setting on a street, diverse distractions have to be considered, this might be different in a more closed environment.

From the case studies, we learned that violating one criterion or several criteria should not necessarily lead to overall rejection of a concept. In contrast, it should be used to improve the concept. However, being unable to fulfill a criterion, even in several iterations, can be a good indicator that there is something wrong with the overall approach of the authentication mechanism. Taking into account the negative influences that not fulfilling specific criteria can have, this can be an important factor.

Therefore, we claim that one of the main lessons learned in this chapter are not that the criteria work as proposed. It is that they can be considered valid tools that support iterative software development approaches. That is, in each iterative step, the criteria are checked and based on the results, the concept, system, software has to be extended or improved. In the next iteration, this has to be checked upon again until the desired level of compliance is reached.

Chapter 7

Conclusion and Future Work

It's not wise to violate rules until you know how to observe them.

– T. S. Eliot –

Secure and usable authentication is a very accessible and grateful sub-field of usable privacy and security. The community working on solutions is rather small and related literature is mainly limited to the last two decades. This means that proposing a solution or having and implementing an idea can be done in a very small fraction of time with a good chance to have the work published as "yet another solution". This is one of the reasons why, as we claimed in this thesis, there is no common ground on which design and evaluation of authentication mechanisms can be based upon. Additionally, most systems are proposed by researchers as a one-time contribution to the field after which they go back to dealing with other usability or security problems.

At the beginning of this work, we identified different problems in the field. There is a big variety of proposed solutions for the authentication problem. However, they do have different disadvantages. Another problem is the lack of a common approach on how authentication mechanisms should be rated and how they should be evaluated. Evaluation approaches are so diverse that it is nearly impossible to compare different solutions in the literature to each other, making it hard to judge which one is better for a specific purpose.

Therefore, in this thesis, we created criteria based on analyses of diverse systems and a large amount of implementations and studies. We could show that fulfilling the criteria comes with benefits while violating them has to be paid with diminished usability and security.

7.1 Summary of the Contributions

The problems we wanted to contribute to with this thesis are to that effect: fix existing problems of usable and secure authentication mechanisms and create guidelines, criteria or a theoretical framework to judge them. The main goal was to be able to improve authentication mechanisms in early design stages and make different systems comparable.

Summarized, we proposed several systems that fix problems of current approaches and related work with respect to the three categories software, hardware and user owned device based approaches as introduced in chapter 2. Furthermore, based on those newly developed authentication mechanisms, long-term as well as field studies, we derived seven criteria, which we could show not only make evaluating and comparing approaches easier but which also can be used to judge and evaluate their quality.

7.1.1 Improving Authentication

Within this thesis, the standard classification of authentication mechanisms has been neglected and instead a new classification has been developed and prioritized as described in chapter 2. This new classification divides authentication mechanisms in software, hardware and user owned device based approaches. The main advantage of this classification lies in the fact that it highlights the advantages and disadvantages of the different systems with respect to a public setting.

One of the main contributions lies in solving parts of these problems by proposing new authentication mechanisms that have been designed to fill these gaps as described in chapter 3. Overall, we proposed five systems that partially build upon each other and iteratively solve the respective problems. In the following, only the best solution for each category will be recapitulated.

The first category, software based solutions, has the advantage that such systems are very easy and cheap to deploy. On the negative side, they usually only provide limited security (only being shoulder surfing resistant) combined with a bad performance with respect to authentication speed. ColorPIN [32] provides advanced security. Even a perfect attack can only reveal the authentication token if it is applied several consecutive times (at least two). Repeated use of the system lead to very good authentication speed around 3.5 seconds. However, an untrained user takes around 14 seconds to authenticate. Even though there are still issues to solve, like partial lack of consistency, we are not aware of a purely software based authentication system that achieves such a high security while remaining as fast and easy to use.

Authentication mechanisms based on hardware owned directly by the user allow for highly secure as well as cheap approaches since the hardware, usually a mobile phone, does not have to be purchased by the service provider. These approaches also provide high security. On the downside, the hardware has to be connected to the terminal, mostly over a wireless channel. This makes them vulnerable to man-in-the-middle attacks. From a usability perspective, connection comes with problems. The connection has to be very secure since in most cases, sensitive data will be transmitted over the wireless channel. This makes establishing a connection slow and sometimes cumbersome. VibraPass [34] has been designed in a way that no sensitive data will be transmitted. The actual input of the PIN still takes place at the terminal's keyboard. The mobile device itself is only used as a feedback channel from the terminal to the user, transmitting messages telling the user to lie at some points. This way, a secure channel is not required and faster connection mechanisms can be employed. The pure authentication speed of VibraPass is very high and the system proved to be highly secure as well. Still, this connection time has to be counted as part of the authentication in case a connected device is not required for the terminal interaction itself which, despite its improvements, can still be considered a possible problem.

Hardware based authentication mechanisms provide high security and good performance. They fail, however, when it comes to resistance to manipulations and deployment costs. Installing advanced hardware at public terminals is still a factor that will hinder their distribution. EyePassShapes [27] is the final result of a long chain of work that we performed over a period of two years [35, 36, 37, 132]. Using PassShapes [132] as authentication tokens, the system utilized a highly memorable and usable graphical approach. At the same time, performing the token with the gaze, the input is very secure and remains fast, especially if the shape is "drawn" in one single attempt. The deployment problem is approached by relying on a gaze-based interaction, gaze gestures [43], that do not require calibration and work with cheap camera hardware. For security reasons, most public terminals are already equipped with cameras, filming the users' faces. Therefore, cheap or no additional hardware at all are required to use the system. Resistance to manipulation is given as well since a successful attack would require the attackers to place their own eye tracking equipment and keypad at the terminal.

Of all the systems, EyePassShapes performed best, both in practical and theoretical analyses. As shown in chapter 6.2, this is in line with fulfilling the criteria identified throughout this thesis.

Even though we could successfully solve many problems of authentication mechanisms for public spaces, the contribution of working on these systems does not only lie in this aspect. Designing, implementing and evaluating them provided in-depth insights on problems and gaps in the design and evaluation process. For instance, the evaluation of VibraPass and MobilePIN provided first hints on the necessity and appropriateness of new criteria. Therefore, it significantly influenced the second contribution of this thesis.

7.1.2 Improving Development (and Evaluation)

One of the main problems of the field of usable and secure authentication mechanisms, within and outside of the public setting, became quickly apparent when analyzing related work. This became even more apparent during the design and evaluation of the authentication mechanisms presented in chapter 3. Even though the majority of related work describes thorough and interesting evaluations, there is no common approach for evaluation and moreover reporting results. This makes comparing different solutions harder than necessary. It is basically impossible to say that system A is better than system B since the information to objectively judge this is not or only partially available. We encountered this problem in its gravity when we analyzed our approaches on a comparative basis.

Therefore, the second main contribution of this thesis is a set of criteria on how to design and evaluate authentication mechanisms with the goal to improve them in a very early design stage and make the outcome comparable. It is not necessary though that a system is superior to others since it fulfills more of the criteria. Depending on the context in which the system is used, some criteria can be considered more important than others. This is to be decided by the respective researcher but it has to be kept in mind that not fulfilling specific criteria can limit the system's security, usability or both.

With respect to a public setting as defined in chapter 1.1, we found that social factors play a very important role and can significantly influence security as well as usability. However, the analysis of related work (see chapter 2.5) revealed that there is only few work on how behavioral factors influence authentication. Not surprisingly, two of the criteria that were defined in this thesis are therefore directly related to social interaction and based on other people being in immediate vicinity to the user.

The work presented in chapter 3 represented the first active step further to identify weaknesses of the standard evaluation process of authentication mechanisms. It helped to improve and newly define criteria that could not have found without implementing and evaluating real prototypes. For instance, the work on MobilePIN [29] showed that when designing and evaluating an authentication mechanism for public spaces, it is extremely important to be honest with measuring authentication speed, and to explicitly define which interaction is part of the authentication process and which is not. We could also show that practical security evaluations often reveal weaknesses that are impossible to find in a theoretical analysis. An example of a new finding is what ended up to become criterion 5: security should not require an active user. This criterion is based on findings of the evaluation of VibraPass [34]. It showed that the theoretical security of the system was negatively influences by so-called "bad lies" by the users. This is a design flaw that we had to learn where it comes from and how to avoid it in the future. Finally, working on authentication mechanisms revealed knowledge gaps that then could be filled in the following work by performing long-term and field studies.

Based on these results, in chapter 4, a long-term study on PIN-entry with different keypad settings is presented. We made the decision to use standard PIN-entry. This way, the users were dealing with a concept they are already very familiar with and thus, novelty effects and the like could be minimized. With the knowledge that time measurement has to be very precise, an authentication speed measurement method based on different phases was proposed and applied: preparation, active authentication and cleanup phase. We could show that this approach has multiple benefits, and that in fact it helped to reveal very important factors that would have stayed hidden otherwise. For instance, a critical disadvantage of random keypad layouts considering the preparation phase could only be revealed applying this advanced measurements. Additionally, the study showed that both, inner and outer consistency are important factors that an authentication mechanisms should fulfill to a certain degree. It gave several hints that randomization has a bad influence on the performance of an authentication mechanisms and should thus be avoided if possible. This is a serious limitation since randomization is often the tool of choice to make a system more secure. These results directly influenced criteria 2 and 3.

Since the long-term study on PIN-entry focused on technical aspects of authentication, the criteria influenced by it are rather technical as well. To deal with the lack of knowledge on behavioral factors, their behavior and their influence on the authentication process, we conducted a long-term field study on ATM use in combination with a follow-up field study and a row of public interviews, presented in chapter 5. Based on this, five implications were derived dealing with authentication speed, memorability issues, social factors, distraction and the users' role in security. These findings influenced criteria 2, 4, 5, 6 and 7 respectively. For instance, we could show that in instances in which memorability was an issue, this fact did not only reduce usability, it also tempered with security. Another important finding was that in public settings, diverse distractions do occur that have a negative effect on security and usability and thus should be taken into account when designing authentication mechanisms.

In chapter 6, we presented a thorough description of the seven final criteria and how they apply to a standard development process. The criteria play different roles in the different phases of the development process. With respect to the contribution of the criteria, it was very important for us to find out, whether fulfilling them or not does actually play a role in practice besides the theoretical benefits that each of them have. Therefore, we applied them to two authentication mechanisms presented in this thesis, VibraPass and EyePassShapes. EyePassShapes, which is the more promising approach of the two, fulfills most of the criteria, while many of them are violated by VibraPass. This way, we could show that applying the criteria not only makes the design process of authentication mechanisms for public spaces more efficient, it also leads to "better" systems.

The criteria were created, extended and improved based on a long row of work covering a range of four years. In their current state, they are highly refined and proved their importance in many different contexts: For instance, the phase based time measurement revealed problems of randomized interfaces that would have stayed hidden measuring plain authentication only. We could also show that memorability does not only influence usability but also security since it can lead to insecure behavior. The field study showed how social factors can hinder security. Such problems can be avoided by adhering to the respective criteria.

Using and applying these criteria will help to refine the concepts with respect to very important factors in very early design stages. Additionally, it provides a way to evaluate authentication mechanisms to reveal very important findings and to directly compare the results to other systems. The seven criteria were created with the context of public spaces. Nevertheless, we argue that they can be applied to any authentication mechanism with the limitation that some of them might be neglected. For instance, designing an authentication system for a private home scenario limits the validity of criterion 6, social compatibility, to family members and close friends. Criteria 1, 2, 3 and 4 however are mostly independent of any context.

7.2 Open and Future Work: Authentication

In this thesis, we presented a catalog of seven criteria. This number is not based on a voluntary decision. It just happened that after four years of intensive work, we ended up having this number of criteria. Each single criterion is not based on a one-time simple observation or thought. They are all the result of ongoing work and thus each criterion has been extended and improved a multitude of times. Therefore, we argue that in the next years, the criteria are likely to be further improved. It is also possible that based on future work, new criteria have to be defined to fill gaps that we are not yet aware of. That is, the catalog as presented in chapter 6 should not be seen as a final truth but rather a helpful toolbox that is likely to be extended with more tools.

Criterion 6 (resistance to distractions) has never been approached in a practical way within this thesis. When applying the criteria to VibraPass and EyePassShapes (see chapter 6.2), it was only tested upon theoretically. The nature of this criterion does actually encourage such an approach since a theoretical analysis does reveal most of what we need to know to judge this factor. For instance, if an authentication attempt has to be fulfilled in a single stroke and will fail otherwise, the criterion can be considered violated. On the other hand, a system like standard PIN-entry is very resistant to distractions because each digit can be input separately without a time limit. Practically, this might be a problem. While theoretically, resuming PIN-entry is no problem, it can limit the use of advanced memorability strategies and thus resuming can fail.

This raises the question how and if an authentication mechanism can be practically tested on resistance to distractions, especially in a lab study. Such a research question is both very difficult to answer and highly interesting at the same time: Can we develop a good method for testing distractions in a lab setting? We can think of a great deal of solutions for this problem but their appropriateness is yet to be proven. A very promising candidate could be an extended version of the cave-like setting as described by Dunphy et al. in [44]. They set up a fake ATM terminal in their lab and surrounded it with a projection of a public setting as shown in figure 7.1. In such a setting, active distractions could be injected to test how users react to them and if they are able to resume the authentication process after the distraction occurred. A distraction could be a passerby asking the user for help or the like. We observed similar and other distractions in our field study as described in chapter 5. One way could therefore be to model these distractions and add them to the study scenario.

Even though technically possible, distractions might be too artificial in a lab setting and are likely not to distract users at all. This is something that has to be evaluated carefully. Only to name a few, important research questions are therefore: To which degree is it possible to distract a user in an artificial setting? How immersive does such a setting have to be to work? What intensity of distraction are required to attract attention? This shows that this problem cannot be simply solved but we believe that it is worth investigating. This way, a methodology could be created to test distractions in lab settings not only for authentication tasks, but for any kind of interaction technology.

It has to be noted that distraction plays an important role also in other research areas. This is however usually approached the other way round. For instance, the lane-change task [92]

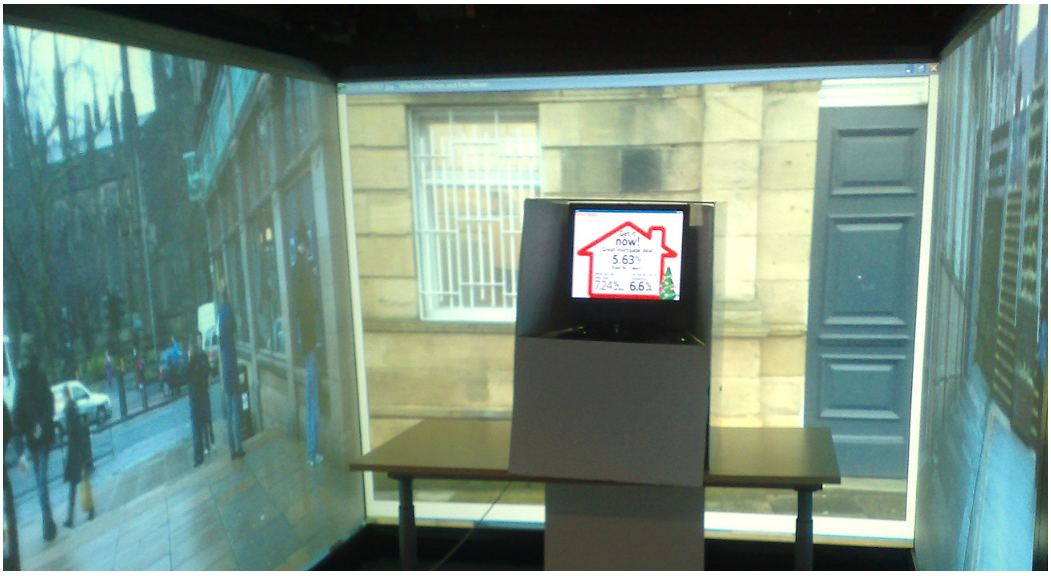


Figure 7.1: Intentional distractions in an ATM lab study. This setup has been used by Dunphy et al. in [44] to provide a more realistic setting for ATM interaction within a controlled lab study. The screens around the ATM mock up display recordings that have been made around a real ATM.

is a methodology designed for the evaluation of in-car interfaces that tests how much (using a quantitative measurement) a secondary task distracts a user from the primary task, which is driving. What would be needed to evaluate criterion 6 would be a methodology to actively distract a user rather than measuring distraction.

We presented significant improvements to current authentication systems but we cannot claim that we created *"the one authentication mechanism to replace them all"*. From our experience based on this work, we argue that there is a high chance that such a system simply does not exist and that it is highly depending on the context. That is, in a different context, a different system might be the best solution.

At this point, we have to ask again whether biometrics might be the solution. From a point of view of the criteria defined in this thesis, biometric authentication in general would be a great candidate. As discussed in chapter 2.1, mainly privacy concerns still hinder its wide adaptation. Therefore, if biometrics are to replace standard authentication, this aspect has to be dealt with both in industry as well as in research. The main problem is the collection and storage of biometric data which users consider lost forever, once it is in possession of another entity, like a bank.

The biometric daemon [13] is a concept describing a possible solution: A pet, that lives with its users and "learns" their biometric features from them. Authentication is done by the daemon rather than its user. Once the user moves too far from it, the daemon dies. Having the biometric data always close to them, this concept might allay the users' doubts. A very important thought of this concept is to create a biometric system in which the data does not have to be given away but

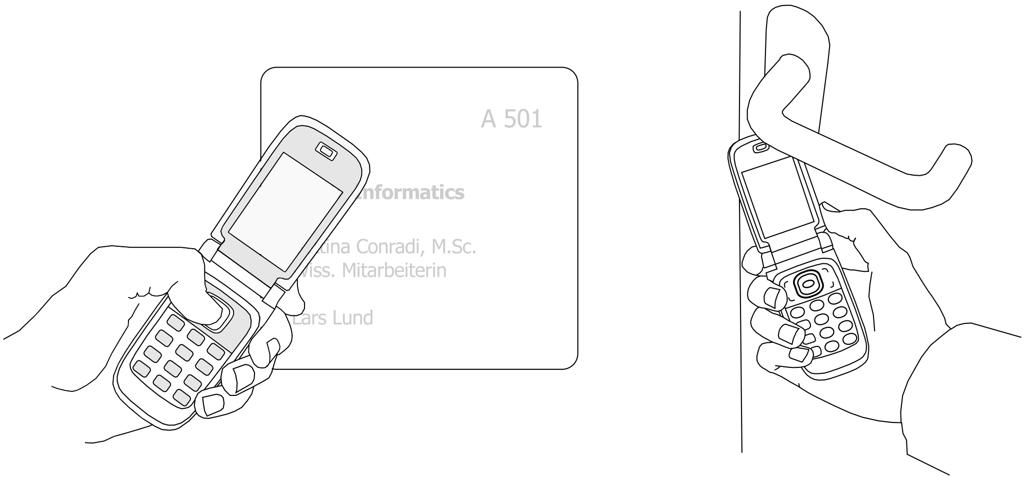


Figure 7.2: Interaction with TreasurePhone [116]. Different profiles of the phone and corresponding data are protected until the user is in the right context. Left: By touching a room plate enhanced with an NFC tag, the user activates a location that defines a context like “work” or “home”. Right: Controlling an NFC-based lock activates a specific action that defines a context as well.

always remains with the user. We can imagine a biometric box (which could be a smartphone for example) or something key-like. The box learns the biometric data from the users while it stays with them. Starting from simple data like fingerprints to biometric information on how the users move, how they hold the box, etc. To authenticate to a system, the box or key has to be “opened” or “made ready”. To do so, biometric features of the user are exploited. In the best case, this could happen implicitly while pulling the box from the pocket. To authenticate to a system, the only property that it has to know is whether the box is open or not. After authentication, the box closes again. Since the biometric data never leaves the device, the users do not have to provide them to a third entity. This is just a quick thought but it highlights that if researchers work on the privacy problem, biometric authentication can be an important factor in our future.

Besides token-based and biometric authentication, there is a third field of authentication that has great potential, implicit authentication. The main idea is that authentication is implicitly happening and not anymore something the user actively does. That is, it eliminates active authentication as a cumbersome task that users do not want to be bothered with since it is not their primary goal [133]. Oftentimes, biometric information is used to achieve this goal, but in many cases, context information can be used as well. When we developed TreasurePhone [116], we created a system in which context is used to define whether a specific profile (and with this specific data) of a smartphone can be viewed or not. Therefore, it uses actions and locations as shown in figure 7.2. For instance, when a user opens an office door with the mobile device (e.g. using NFC), the phone switches to the “work” location and data related to the user’s work becomes available. Even though this is authorization rather than authentication, the system shows how context can be used to grant or deny access to a specific entity.

Especially the field of mobile personal devices can highly benefit from implicit authentication. For instance, modern mobile phones have different mechanisms to unlock the displays or keypads so that the user can interact with the device. These include for instance moving locks from one side to another or dragging windows down. Adding a biometric component to this approach, the mobile device could not only measure that the unlock mechanism was used but also how. This way, the user can be identified and authorized to use the phone or not. That is, authentication can take place implicitly in such an approach. Other approaches of implicit authentication could include shaking patterns and the like. Theoretically, the criteria of implicit authentication do not differ significantly from the ones presented in this thesis. Even though authentication is happening implicitly, the design of the system is important for its speed, security, robustness to distractions etc.[1]

In this chapter, we showed that there is still a large body of open research questions related to authentication. We are currently continuing our work with a focus on the just presented topics.

7.3 Open and Future Work: Internet Security

Usable privacy and security is not limited to authentication. The field is huge and any interaction that involves private, critical or sensitive data is a possible field of interest for researchers. Working in this field, one cannot refrain from looking left and right. On both sides, there is amazing work to see and big gaps to fill. In this chapter, we will talk about one of these topics, one that currently raised our interest and will be amongst the major focuses of our future work.

One of the main lessons that can be learned from usable and secure authentication is that it does not matter how thorough and impeccable the security of an authentication mechanism is if users simply give away the data that was meant to be protected. Among the criteria, number 5 (security should not require an active user) can be considered the last line of defense against insecure behavior. That is, when properly implemented, it protects users from revealing their authentication token due to "inappropriate" behavior. However, there is no protection for users that voluntarily and actively give away their authentication tokens or any other private data in the believe that the receiving entity is trustworthy. This, in fact, is a very common problem of Internet use.

Securing Internet users from frauds is maybe the most attended to topic in usable privacy and security. Additionally, it has an extremely wide coverage in the media. We are almost daily confronted with news about new Phishing attacks or other frauds meant to steal credentials while using the Internet that can be used later on to cause significant (financial) damage. All approaches have one typical property in common: They try to gain the users' trust by counterfeiting trusted services or creating trust by other means. For instance, a study by Fogg et al. [53] showed that the "look and feel", meaning the design of a website, is more likely to create trust than any

[1] The author of this thesis recently received a Google Research Award funding to conduct a project on implicit authentication on mobile devices.

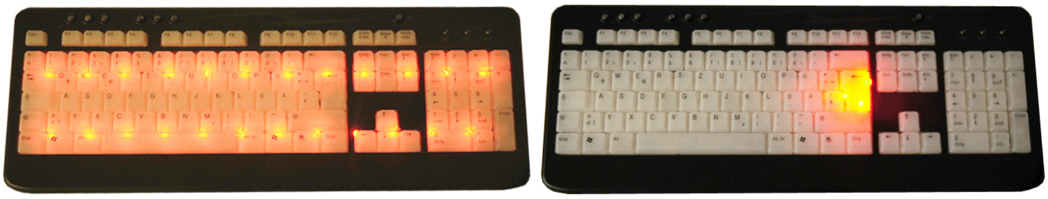


Figure 7.3: Ambient security visualization using MoodyBoard [30, 31]. Left: The whole keyboard is glowing in red to signalize dangerous behavior like inputting a credit card number on a website without SSL certificate. Right: The return key can be lit separately to enrich the feedback with a deeper meaning like “if you submit this data, it might be stolen”.

other factor. Objectively and technically, this is a property that does not have any influence on the security of a system. That is that in a subjectively trusted situation, users easily think that providing their personal data, like credit card information, is not risky.

Current solutions for protecting Internet users from frauds are either blocking – forcing the users to decide upon one out of a set of given actions [46] – or non-blocking. The latter leave it to the user to check warnings or not [138]. SSL-certificate visualizations of current browsers fall into this category. The third and final approach is using teaching mechanism to train the users to behave more securely and how to identify threats [80, 119]. Reasons why such mechanisms fail are manifold. Habituation effects are an often cited problem that describe a situation in which users mix up important warnings with unimportant warnings they are often confronted with [3]. Overlooking warnings [138], plain lack of interest in security [133], lack of required knowledge [123] and wrong mental models [41] are another few that have to be mentioned.

With ambient security visualization, we are currently conducting work on a fourth approach which can be seen as a non-blocking system. As opposed to them, by using ambient information rather than graphical user interfaces, such a system does not occupy any screen real estate. At the same time, very strong (or intense) notifications can be used that are less likely to be overseen by a user. We are not aware of any related work that employs ambient information to transport privacy and security relevant information or warnings to a user. Therefore, this can be considered a new subfield of Internet security research. First results indicate that ambient visualization has various advantages compared to GUI-based security warnings.

We started the experiments in this area using a keyboard to transmit vibration and color-coded information to a user while browsing the web. First steps including field studies and theoretical analyses to figure out an optimal (or at least good) configuration for such a system [30]. We also conducted user studies conducted with a version of MoodyBoard that only warned users about problems using both, vibration and colors as shown in figure 7.3. These studies showed that the system achieved security awareness comparable to blocking warnings while keeping the advantages of non-blocking systems: not interrupting the current task of the user and occupying limited or no screen real estate [31].

Besides effectiveness, another big advantage of ambient security visualization can be seen in its consistency even when transferred to completely different contexts. While GUI-based approaches often have the problem that they have been designed for a specific physical setup (for instance a specific minimum screen resolution), ambient security visualization is completely independent from such limitations. The metaphor of a red warning glowing works on a desktop environment as well as in a mobile setting without loosing any of its meaning. For instance in [93], Maurer discusses how ambient security visualization can be used in a mobile context.

7.4 One Final Outlook

With the criteria, we created a basis to support researchers and anyone working on usable and secure authentication mechanisms for public spaces. The criteria are helpful in many aspects. They simplify the design process and enable rejecting or improving concepts in very early development phases. Thus, the can help saving both, money and effort. Additionally, they "force" developers to be very accurate. As opposed to other fields of human-computer interaction, we are mainly dealing with extremely focused and short tasks. To make this even worse, we have to handle situations and tasks which the users do not want to be bothered with since – to state it one final time – they are not their primary goals. Therefore, usable security systems, such as authentication, need to be very efficient. That is, the sensation of added security should, if not be joyful, at least be no burden to the users. To achieve this, accuracy or precision in evaluating the systems is of high importance. The criteria presented in this thesis help to achieve accuracy by being very precise with how to measure and evaluate prototypes and ideas.

Finally, if the criteria are widely applied, it would actually be possible to easily compare different solutions to each other. This way, it would be easy to judge the appropriateness of respective mechanisms. Even more, it would allow for ranking the systems and making informed decisions on which one will work better for a specific task. That said, we argue that in this thesis, we provide a solution for the evaluation problem of current secure authentication mechanisms. Now it is on others to decide whether this effort will be fruitful or not.

BIBLIOGRAPHY

[1] Abrams, M. D. Security engineering in an evolutionary acquisition environment. In *NSPW '98: Proceedings of the 1998 workshop on New security paradigms*, ACM, 1998, pp. 11–20.

[2] Adams, A. and Sasse, M. A. Users are not the enemy. *Commun. ACM 42, 12* (1999), 40–46.

[3] Amer, T. and Maris, J. Signal words and signal icons in application control and information technology exception messages–hazard matching and habituation effects. *Journal of Information Systems 21, 2* (2006).

[4] Ashbourn, J. *Biometrics: advanced identity verification*. Springer-Verlag, London, UK, 2000.

[5] Bauer, L., Cranor, L. F., Reiter, M. K., and Vaniea, K. Lessons learned from the deployment of a smartphone-based access-control system. In *SOUPS '07: Proceedings of the 3rd symposium on Usable privacy and security*, ACM, 2007, pp. 64–75.

[6] Bianchi, A., Oakley, I., Kostakos, V., and Kwon, D. S. The phone lock: Audio and haptic shoulder-surfing resistant pin entry methods. In *TEI '11: Proceedings of the 5th international conference on Tangible, Embedded and Embodied Interaction*, ACM, 2011.

[7] Bianchi, A., Oakley, I., and Kwon, D. S. The secure haptic keypad: a tactile password system. In *CHI '10: Proceedings of the 28th international conference on Human factors in computing systems*, ACM, 2010, pp. 1089–1092.

[8] Bianchi, A., Oakley, I., Lee, J. K., and Kwon, D. S. The haptic wheel: design and evaluation of a tactile password system. In *CHI EA '10: Proceedings of the 28th international conference extended abstracts on Human factors in computing systems*, ACM, 2010, pp. 3625–3630.

[9] Boring, S. and Baur, D. Can you see where i point at? In *2nd International Workshop on Security and Privacy in Spontaneous Interaction and Mobile Phone Use (in Conjunction with Pervasive 2010), Helsinki, Finland, May 2010*, May 2010.

[10] Boring, S., Baur, D., Butz, A., Gustafson, S., and Baudisch, P. Touch projector: mobile interaction through video. In *CHI '10: Proceedings of the 28th international conference on Human factors in computing systems*, ACM, 2010, pp. 2287–2296.

[11] Brainard, J., Juels, A., Rivest, R. L., Szydlo, M., and Yung, M. Fourth-factor authentication: somebody you know. In *CCS '06: Proceedings of the 13th ACM conference on Computer and communications security*, ACM, 2006, pp. 168–178.

[12] Brewster, S. and Brown, L. M. Tactons: structured tactile messages for non-visual information display. In *AUIC '04: Proceedings of the fifth conference on Australasian user interface*, Australian Computer Society, Inc., 2004, pp. 15–23.

[13] Briggs, P. and Olivier, P. L. Biometric daemons: authentication via electronic pets. In *CHI '08 extended abstracts on Human factors in computing systems*, CHI '08, ACM, 2008, pp. 2423–2432.

[14] Bulling, A., Roggen, D., and Tröster, G. It's in your eyes - towards context-awareness and mobile hci using wearable eog goggles. In *Proc. of the 10th International Conference on Ubiquitous Computing (UbiComp 2008)*, volume 344 of *ACM International Conference Proceeding Series*, ACM Press, Sept. 2008, pp. 84–93.

[15] Carr, S. *Public Space*. Cambridge Univ Pr, 1992.

[16] Chiasson, S., Biddle, R., and van Oorschot, P. C. A second look at the usability of click-based graphical passwords. In *SOUPS '07: Proceedings of the 3rd symposium on Usable privacy and security*, ACM, 2007, pp. 1–12.

[17] Chiasson, S., Van Oorschot, P. C., and Biddle, R. Graphical password authentication using cued click points. In *12 th European Symposium On Research In Computer Security (ESORICS), 2007*, Springer-Verlag, 2007.

[18] Chong, M. and Marsden, G. Exploring the use of discrete gestures for authentication. In *Human Computer Interaction - INTERACT 2009*, volume 5727, Springer Berlin Heidelberg, 2009, pp. 205 – 213.

[19] Claycomb, W. and Shin, D. Using a two dimensional colorized barcode solution for authentication in pervasive computing. In *PERSER '06: Proceedings of the 2006 ACS/IEEE International Conference on Pervasive Services*, IEEE Computer Society, 2006, pp. 173–180.

[20] Claycomb, W. and Shin, D. Towards secure resource sharing for impromptu collaboration in pervasive computing. In *SAC '07: Proceedings of the 2007 ACM symposium on Applied computing*, ACM, 2007, pp. 940–946.

[21] Conrad, R. Short-term memory effects in the design of data-entry keyboards. *Journal of Applied Psychology 50, 5* (10 1966), 353–356.

[22] Conrad, R. and Hull, A. J. The preferred layout for numeral data-entry keysets. *Ergonomics 11, 2* (1968), 165–173.

[23] Coventry, L., De Angeli, A., and Johnson, G. Usability and biometric verification at the atm interface. In *CHI '03: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, 2003, pp. 153–160.

[24] Davis, D., Monrose, F., and Reiter, M. K. On user choice in graphical password schemes. In *SSYM'04: Proceedings of the 13th conference on USENIX Security Symposium*, USENIX Association, 2004, pp. 11–11.

[25] De Angeli, A., Coutts, M., Coventry, L., Johnson, G. I., Cameron, D., and Fischer, M. H. Vip: a visual approach to user authentication. In *AVI '02: Proceedings of the Working Conference on Advanced Visual Interfaces*, ACM, 2002, pp. 316–323.

[26] De Angeli, A., Coventry, L., Johnson, G., and Renaud, K. Is a picture really worth a thousand words? exploring the feasibility of graphical authentication systems. *Int. J. Hum.-Comput. Stud. 63, 1-2* (2005), 128–152.

[27] De Luca, A., Denzel, M., and Hussmann, H. Look into my eyes!: can you guess my password? In *SOUPS '09: Proceedings of the 5th Symposium on Usable Privacy and Security*, ACM, 2009, pp. 1–12.

[28] De Luca, A. and Frauendienst, B. A privacy-respectful input method for public terminals. In *NordiCHI '08: Proceedings of the 5th Nordic conference on Human-computer interaction*, ACM, 2008, pp. 455–458.

[29] De Luca, A., Frauendienst, B., Boring, S., and Hussmann, H. My phone is my keypad: privacy-enhanced pin-entry on public terminals. In *OZCHI '09: Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group*, ACM, 2009, pp. 401–404.

[30] De Luca, A., Frauendienst, B., Maurer, M., and Hausen, D. On the design of a "moody" keyboard. In *DIS '10: Proceedings of the 8th ACM Conference on Designing Interactive Systems*, ACM, 2010, pp. 236–239.

[31] De Luca, A., Frauendienst, B., Maurer, M.-E., Seifert, J., Hausen, D., Kammerer, N., and Hussmann, H. Does moodyboard make internet use more secure? evaluating an ambient security visualization tool. In *CHI '11: Proceedings of the 29th international conference on Human factors in computing systems*, ACM, 2011.

[32] De Luca, A., Hertzschuch, K., and Hussmann, H. Colorpin: securing pin entry through indirect input. In *CHI '10: Proceedings of the 28th international conference on Human factors in computing systems*, ACM, 2010, pp. 1103–1106.

[33] De Luca, A., Langheinrich, M., and Hussmann, H. Towards understanding atm security: a field study of real world atm use. In *SOUPS '10: Proceedings of the Sixth Symposium on Usable Privacy and Security*, ACM, 2010, pp. 1–10.

[34] De Luca, A., von Zezschwitz, E., and Hussmann, H. Vibrapass: secure authentication based on shared lies. In *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, ACM, 2009, pp. 913–916.

[35] De Luca, A., Weiss, R., and Drewes, H. Evaluation of eye-gaze interaction methods for security enhanced pin-entry. In *OZCHI '07: Proceedings of the 19th Australasian conference on Computer-Human Interaction*, ACM, 2007, pp. 199–202.

[36] De Luca, A., Weiss, R., and Hussmann, H. Passshape: stroke based shape passwords. In *OZCHI '07: Proceedings of the 19th Australasian conference on Computer-Human Interaction*, ACM, 2007, pp. 239–240.

[37] De Luca, A., Weiss, R., Hussmann, H., and An, X. Eyepass - eye-stroke authentication for public terminals. In *CHI '08: CHI '08 extended abstracts on Human factors in computing systems*, ACM, 2008, pp. 3003–3008.

[38] Deininger, R. L. Human factors engineering studies of the design and use of pushbutton telephone sets. *The Bell System, Technical Journal 4* (1960), 995–1012.

[39] Deyle, T. and Roth, V. Accessible authentication via tactile pin entry. *Computer Graphics Topics Issue 3* (Mar. 2006).

[40] Dhamija, R. and Perrig, A. Déjà vu: a user study using images for authentication. In *SSYM'00: Proceedings of the 9th conference on USENIX Security Symposium*, USENIX Association, 2000, pp. 4–4.

[41] Dhamija, R., Tygar, J. D., and Hearst, M. Why phishing works. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, CHI '06, ACM, 2006, pp. 581–590.

[42] Dirik, A. E., Memon, N., and Birget, J.-C. Modeling user choice in the passpoints graphical password scheme. In *SOUPS '07: Proceedings of the 3rd symposium on Usable privacy and security*, ACM, 2007, pp. 20–28.

[43] Drewes, H. and Schmidt, A. Interacting with the computer using gaze gestures. In *INTERACT'07: Proceedings of the 11th IFIP TC 13 international conference on Human-computer interaction*, Springer-Verlag, 2007, pp. 475–488.

[44] Dunphy, P., Fitch, A., and Olivier, P. Gaze-contingent passwords at the atm. In *COGAIN '08: Proceedings of the 4th Conference on Communication by Gaze Interaction - Communication, Environment and Mobility Control by Gaze*, 2008.

[45] Dunphy, P. and Yan, J. Do background images improve "draw a secret" graphical passwords? In *CCS '07: Proceedings of the 14th ACM conference on Computer and communications security*, ACM, 2007, pp. 36–47.

[46] Egelman, S., Cranor, L. F., and Hong, J. You've been warned: an empirical study of the effectiveness of web browser phishing warnings. In *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, CHI '08, ACM, 2008, pp. 1065–1074.

[47] Emilien, G., Durlach, C., Antoniadis, E., van der Linden, M., and Maloteaux, J. *Memory: neuropsychological, imaging, and psychopharmacological perspectives*. Psychology Press, 2004.

[48] Everitt, K. M., Bragin, T., Fogarty, J., and Kohno, T. A comprehensive study of frequency, interference, and training of multiple graphical passwords. In *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, ACM, 2009, pp. 889–898.

[49] Flechais, I., Mascolo, C., and Sasse, M. A. Integrating security and usability into the requirements and design process. In *Second International Conference on Global E-Security*, 2006.

[50] Flechais, I., Sasse, M. A., and Hailes, S. M. V. Bringing security home: a process for developing secure and usable systems. In *NSPW '03: Proceedings of the 2003 workshop on New security paradigms*, ACM, 2003, pp. 49–57.

[51] Fleishman, E. and Parker, J. Factors in the retention and relearning of perceptual-motor skill. *Journal of Experimental Psychology 64* (1962), 215–226.

[52] Florêncio, D. and Herley, C. Where do security policies come from? In *SOUPS '10: Proceedings of the Sixth Symposium on Usable Privacy and Security*, ACM, 2010, pp. 1–14.

[53] Fogg, B. J., Marshall, J., Laraki, O., Osipovich, A., Varma, C., Fang, N., Paul, J., Rangnekar, A., Shon, J., Swani, P., and Treinen, M. What makes web sites credible?: a report on a large quantitative study. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '01, ACM, 2001, pp. 61–68.

[54] Forget, A., Chiasson, S., and Biddle, R. Shoulder-surfing resistance with eye-gaze entry in cued-recall graphical passwords. In *CHI '10: Proceedings of the 28th international conference on Human factors in computing systems*, ACM, 2010, pp. 1107–1110.

[55] Goffman, E. *The presentation of self in everyday life*. Anchor Books, May 1959.

[56] Goodman, D., Dickinson, J., and Francas, M. J. Human factors design considerations for public videotex input devices. *Behaviour & Information Technology 4, 3* (1985), 189–200.

[57] Grudin, J. The case against user interface consistency. *Commun. ACM 32, 10* (1989), 1164–1173.

[58] Hatta, K. and Liyama, Y. Ergonomic study of automatic teller machine operability. *International Journal of Human-Computer Interaction 3, 3* (1991), 295–309.

[59] Hayashi, E., Dhamija, R., Christin, N., and Perrig, A. Use your illusion: secure authentication usable anywhere. In *SOUPS '08: Proceedings of the 4th symposium on Usable privacy and security*, ACM, 2008, pp. 35–45.

[60] Higbee, K. L. *Your Memory : How It Works and How to Improve It.* Da Capo Press, Mar. 2001.

[61] Hoanca, B. and Mock, K. Secure graphical password system for high traffic public areas. In *ETRA '06: Proceedings of the 2006 symposium on Eye tracking research & applications*, ACM, 2006, pp. 35–35.

[62] Houmb, S., Islam, S., Knauss, E., Jürjens, J., and Schneider, K. Eliciting security requirements and tracing them to design: an integration of common criteria, heuristics, and umlsec. *Requirements Engineering 15* (2010), 63–93.

[63] Huang, E. M., Koster, A., and Borchers, J. Overcoming assumptions and uncovering practices: When does the public really look at public displays?. In Indulska, J., Patterson, D. J., Rodden, T., and Ott, M., editors, *Pervasive*, volume 5013 of *Lecture Notes in Computer Science*, Springer, 2008, pp. 228–243.

[64] Imanaka, K., Yamauchi, M., Funase, K., and Nishihira, Y. Information-processing mediating the location-distance interference in motor short-term memory. *The Annals of physiolocial anthropology 5* (1993), 269–283.

[65] Jackson, C., Simon, D. R., Tan, D. S., and Barth, A. An evaluation of extended validation and picture-in-picture phishing attacks. In *USEC '07: Proceedings of Usable Security*, 2007.

[66] Jacob, R. J. What you look at is what you get: eye movement-based interaction techniques. In *CHI '90: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, 1990, pp. 11–18.

[67] Jermyn, I., Mayer, A., Monrose, F., Reiter, M. K., and Rubin, A. D. The design and analysis of graphical passwords. In *SSYM'99: Proceedings of the 8th conference on USENIX Security Symposium*, USENIX Association, 1999, pp. 1–1.

[68] Johnson, K. and Werner, S. Using composite scene authentication (csa) as a graphical alternative to alphanumeric password systems. *Human Factors and Ergonomics Society Annual Meeting Proceedings 50* (2006), 661–664.

[69] Johnson, K. and Werner, S. Memorability of alphanumeric and composite scene authentication (csa) passcodes over extended retention intervals. *Human Factors and Ergonomics Society Annual Meeting Proceedings 51* (2007), 434–438.

[70] Johnson, K. and Werner, S. Graphical user authentication: A comparative evaluation of composite scene authentication vs. three competing graphical passcode systems. *Human Factors and Ergonomics Society Annual Meeting Proceedings 52* (2008), 542–546.

[71] Jürjens, J. Umlsec: Extending uml for secure systems development. In Jezequel, J.-M., Hussmann, H., and Cook, S., editors, *UML 2002 - The Unified Modeling Language*, volume 2460 of *Lecture Notes in Computer Science*, pp. 1–9. Springer Berlin / Heidelberg, 2002.

[72] Karn, K. S., Ellis, S., and Juliano, C. The hunt for usability: tracking eye movements. In *CHI '99: CHI '99 extended abstracts on Human factors in computing systems*, ACM, 1999, pp. 173–173.

[73] Kern, D., Marshall, P., and Schmidt, A. Gazemarks: gaze-based visual placeholders to ease attention switching. In *CHI '10: Proceedings of the 28th international conference on Human factors in computing systems*, ACM, 2010, pp. 2093–2102.

[74] Kim, D., Dunphy, P., Briggs, P., Hook, J., Nicholson, J., Nicholson, J., and Olivier, P. Multi-touch authentication on tabletops. In *CHI '10: Proceedings of the 28th international conference on Human factors in computing systems*, ACM, 2010, pp. 1093–1102.

[75] Kirschnick, N., Kratz, S., and Möller, S. An improved approach to gesture-based authentication for mobile devices. In *6th Symposium on Usable Privacy and Security (SOUPS), Redmond, WA*, 2010.

[76] Klein, D. V. Foiling the cracker: A survey of, and improvements to, password security. In *Proceedings of the 2nd USENIX UNIX Security Workshop*, 1990.

[77] Kray, C., Kortuem, G., and Krüger, A. Adaptive navigation support with public displays. In *IUI '05: Proceedings of the 10th international conference on Intelligent user interfaces*, ACM, 2005, pp. 326–328.

[78] Kumar, M., Garfinkel, T., Boneh, D., and Winograd, T. Reducing shoulder-surfing by using gaze-based password entry. In *SOUPS '07: Proceedings of the 3rd symposium on Usable privacy and security*, ACM, 2007, pp. 13–19.

[79] Kumaraguru, P., Cranshaw, J., Acquisti, A., Cranor, L., Hong, J., Blair, M. A., and Pham, T. School of phish: a real-world evaluation of anti-phishing training. In *SOUPS '09: Proceedings of the 5th Symposium on Usable Privacy and Security*, ACM, 2009, pp. 1–12.

[80] Kumaraguru, P., Rhee, Y., Acquisti, A., Cranor, L. F., Hong, J., and Nunge, E. Protecting people from phishing: the design and evaluation of an embedded training email system. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '07, ACM, 2007, pp. 905–914.

[81] Lawson, B. *The language of space*. Architectural Press, Dec. 2001.

[82] LeBlanc, D., Chiasson, S., Forget, A., and Biddle, R. An improved approach to gesture-based authentication for mobile devices. In *4th Symposium on Usable Privacy and Security (SOUPS), Redmond, WA*, 2008.

[83] Little, L. Attitudes towards technology use in public zones: the influence of external factors on atm use. In *CHI '03: CHI '03 extended abstracts on Human factors in computing systems*, ACM, 2003, pp. 990–991.

[84] Little, L., Briggs, P., and Coventry, L. An activity theory approach to technology use in public areas: The case of the atm. In *HCi 2003: Designing for Society. Proceedings of the 17th British HCI Group Annual Conference*, 2003.

[85] Lodderstedt, T., Basin, D., and Doser, J. Secureuml: A uml-based modeling language for model-driven security. In Jezequel, J.-M., Hussmann, H., and Cook, S., editors, *UML 2002 - The Unified Modeling Language*, volume 2460 of *Lecture Notes in Computer Science*, pp. 426–441. Springer Berlin / Heidelberg, 2002.

[86] Lutz, M. C. and Chapanis, A. Expected locations of digits and letters on ten-button keysets. *Journal of Applied Psychology 39, 5* (1955), 314–317.

[87] Majaranta, P., Aula, A., and Räihä, K.-J. Effects of feedback on eye typing with a short dwell time. In *ETRA '04: Proceedings of the 2004 symposium on Eye tracking research & applications*, ACM, 2004, pp. 139–146.

[88] Majaranta, P. and Räihä, K.-J. Twenty years of eye typing: systems and design issues. In *ETRA '02: Proceedings of the 2002 symposium on Eye tracking research & applications*, ACM, 2002, pp. 15–22.

[89] Malek, B., Orozco, M., and El Saddik, A. Novel shoulder-surfing resistant haptic-based graphical password. In *EuroHaptics 2006*, July 2006.

[90] Manzke, J. M., Egan, D. H., Felix, D., and Krueger, H. What makes an automated teller machine usable by blind users? *Ergonomics 41* (1998), 982–999.

[91] Marteniuk, R. G., Ivens, C. J., and Brown, B. E. Are there task specific performance effects for differently configured numeric keypads? *Applied Ergonomics 27, 5* (Oct. 1996), 321–325.

[92] Mattes, S. The lane change task as a tool for driver distraction evaluation. In *Proceedings of IGfA 2003*, 2003.

[93] Maurer, M. Bringing effective security warnings to mobile browsing. In *2nd International Workshop on Security and Privacy in Spontaneous Interaction and Mobile Phone Use (in Conjunction with Pervasive 2010), Helsinki, Finland, May 2010*, May 2010.

[94] Maurer, M.-E. and De Luca, A. Secuui: Autocomplete your terminal input. In *MobileHCI '09: Extended abstract of the 11th international conference on Human computer interaction with mobile devices and services*, 2009.

[95] Miller, K. F., Smith, C. M., Zhu, J., and Zhang, H. Preschool origins of cross-national differences in mathematical competence: The role of number-naming systems. *Psychological Science 6, 1* (1995), 55–60.

[96] Moncur, W. and Leplâtre, G. Pictures at the atm: exploring the usability of multiple graphical passwords. In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, 2007, pp. 887–894.

[97] Nali, D. and Thorpe, J. Analyzing user choice in graphical passwords. Technical report, School of Computer Science, Carleton University, 2004.

[98] Nelson, D. L., Reed, V. S., and Walling, J. R. Pictorial superiority effect. *Journal of Experimental Psychology: Human Learning and Memory 2, 5* (Sept. 1976), 523–528.

[99] Norman, D. *The Design of Everyday Things*. Perseus Books, Aug. 2002.

[100] Peltonen, P., Kurvinen, E., Salovaara, A., Jacucci, G., Ilmonen, T., Evans, J., Oulasvirta, A., and Saarikko, P. It's mine, don't touch!: interactions at a large multi-touch display in a city centre. In *CHI '08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, ACM, 2008, pp. 1285–1294.

[101] Pering, T., Anokwa, Y., and Want, R. Gesture connect: facilitating tangible interaction with a flick of the wrist. In *TEI '07: Proceedings of the 1st international conference on Tangible and embedded interaction*, ACM, 2007, pp. 259–262.

[102] Pinkas, B. and Sander, T. Securing passwords against dictionary attacks. In *CCS '02: Proceedings of the 9th ACM conference on Computer and communications security*, ACM, 2002, pp. 161–170.

[103] Pons, A. P. and Polak, P. Understanding user perspectives on biometric technology. *Commun. ACM 51, 9* (2008), 115–118.

[104] Rogers, J. Please enter your 4-digit pin. *Financial Services Technology, U.S. Edition Issue 4* (Mar. 2007).

[105] Rogers, W. A., Gilbert, D. K., and Cabrera, E. F. An analysis of automatic teller machine usage by older adults: A structured interview approach. *Applied Ergonomics 28, 3* (June 1997), 173–180.

[106] Roth, V., Richter, K., and Freidinger, R. A pin-entry method resilient against shoulder surfing. In *CCS '04: Proceedings of the 11th ACM conference on Computer and communications security*, ACM, 2004, pp. 236–245.

[107] Royal National Institute for the Blind and Gill, J. Access prohibited? information for designers of public access terminals, 1997.

[108] Rukzio, E., Müller, M., and Hardy, R. Design, implementation and evaluation of a novel public display for pedestrian navigation: the rotating compass. In *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, ACM, 2009, pp. 113–122.

[109] Ryu, Y. S., Koh, D. H., Aday, B. L., Gutierrez, X. A., and Platt, J. D. Usability evaluation of randomized keypads. *Journal of Usability Studies 5* (2010), 65–75.

[110] Sasamoto, H., Christin, N., and Hayashi, E. Undercover: authentication usable in front of prying eyes. In *CHI '08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, ACM, 2008, pp. 183–192.

[111] Sasse, M. A., Brostoff, S., and Weirich, D. Transforming the 'weakest link' - a human/computer interaction approach to usable and effective security. *BT Technology Journal 19* (2001), 122–131. 10.1023/A:1011902718709.

[112] Satzinger, J. W. and Olfman, L. User interface consistency across end-user applications: the effects on mental models. *J. Manage. Inf. Syst. 14, 4* (1998), 167–193.

[113] Schechter, S., Egelman, S., and Reeder, R. W. It's not what you know, but who you know: a social approach to last-resort authentication. In *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, ACM, 2009, pp. 1983–1992.

[114] Schmidt, D., Chong, M. K., and Gellersen, H. Handsdown: Hand-contour-based user identification for interactive surfaces. In *NordiCHI '10: Proceedings of the 6th Nordic conference on Human-computer interaction*, 2010.

[115] Schneier, B. Inside risks: the uses and abuses of biometrics. *Commun. ACM 42, 8* (1999), 136.

[116] Seifert, J., De Luca, A., Conradi, B., and Hussmann, H. TreasurePhone: Context-Sensitive user data protection on mobile phones. In *Pervasive 2010*, volume Volume 6030/2010 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2010, pp. 130–137.

[117] Shadmehr, R. and Brashers-krug, T. Functional stages in the formation of human long-term motor memory. *The Journal of Neuroscience 17* (1997), 409–419.

[118] Sharp, R., Scott, J., and Beresford, A. Secure mobile computing via public terminals. In *Pervasive Computing*, 2006, pp. 238–253.

[119] Sheng, S., Magnien, B., Kumaraguru, P., Acquisti, A., Cranor, L. F., Hong, J., and Nunge, E. Anti-phishing phil: the design and evaluation of a game that teaches people not to fall for phish. In *SOUPS '07: Proceedings of the 3rd symposium on Usable privacy and security*, ACM, 2007, pp. 88–99.

[120] Shirazi, A. S., Döring, T., Parvahan, P., Ahrens, B., and Schmidt, A. Poker surface: combining a multi-touch table and mobile phones in interactive card games. In *MobileHCI '09: Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*, ACM, 2009, pp. 1–2.

[121] Simons, D. J. Current approaches to change blindness. *Visual Cognition 7, 1* (2000), 1–15.

[122] Standing, L. Learning 10,000 pictures. *The Quarterly Journal of Experimental Psychology 25* (1973), 203–222.

[123] Sunshine, J., Egelman, S., Almuhimedi, H., Atri, N., and Cranor, L. F. Crying wolf: an empirical study of ssl warning effectiveness. In *Proceedings of the 18th conference on USENIX security symposium*, SSYM'09, USENIX Association, 2009, pp. 399–416.

[124] Tan, D. S., Keyani, P., and Czerwinski, M. Spy-resistant keyboard: more secure password entry on public touch screen displays. In *OZCHI '05: Proceedings of the 19th conference of the computer-human interaction special interest group (CHISIG) of Australia on Computer-human interaction*, Computer-Human Interaction Special Interest Group (CHISIG) of Australia, 2005, pp. 1–10.

[125] Thorpe, J. and van Oorschot, P. C. Graphical dictionaries and the memorable space of graphical passwords. In *SSYM'04: Proceedings of the 13th conference on USENIX Security Symposium*, USENIX Association, 2004, pp. 10–10.

[126] Thorpe, J. and van Oorschot, P. C. Human-seeded attacks and exploiting hot-spots in graphical passwords. In *SS'07: Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium*, USENIX Association, 2007, pp. 1–16.

[127] Thorpe, J., van Oorschot, P. C., and Somayaji, A. Pass-thoughts: authenticating with our minds. In *NSPW '05: Proceedings of the 2005 workshop on New security paradigms*, ACM, 2005, pp. 45–56.

[128] Thrower, K. R. Access control apparatus. US Patent 4,857,917, United States Patent and Trademark Office, Old Cedar, 12 Wychcotes, Caversham, Reading, RG4 7DA, GB2, August 1989.

[129] Vajk, T., Coulton, P., Bamford, W., and Edwards, R. Using a mobile phone as a wii-like controller for playing games on a large public display. *Int. J. Comput. Games Technol. 2008* (2008), 1–6.

[130] von Zezschwitz, E. An evaluation of the influence of external factors on authentication performance and memorability. Diploma Thesis. Media Informatics Group, Ludwig-Maximilians-Universität München (2010).

[131] Weiser, M. The computer for the twenty-first century. *Scientific American 265, 3* (1991), 94–104.

[132] Weiss, R. and De Luca, A. Passshapes: utilizing stroke based authentication to increase password memorability. In *NordiCHI '08: Proceedings of the 5th Nordic conference on Human-computer interaction*, ACM, 2008, pp. 383–392.

[133] Whitten, A. and Tygar, J. D. Why johnny can't encrypt: a usability evaluation of pgp 5.0. In *SSYM'99: Proceedings of the 8th conference on USENIX Security Symposium*, USENIX Association, 1999, pp. 14–14.

[134] Wiedenbeck, S., Waters, J., Birget, J., Brodskiy, A., and Memon, N. PassPoints: design and longitudinal evaluation of a graphical password system. *International Journal of Human-Computer Studies 63, 1-2* (July 2005), 102–127.

[135] Wiedenbeck, S., Waters, J., Sobrado, L., and Birget, J.-C. Design and evaluation of a shoulder-surfing resistant graphical password scheme. In *AVI '06: Proceedings of the working conference on Advanced visual interfaces*, ACM, 2006, pp. 177–184.

[136] Wobbrock, J. O., Myers, B. A., and Kembel, J. A. Edgewrite: a stylus-based text entry method designed for high accuracy and stability of motion. In *UIST '03: Proceedings of the 16th annual ACM symposium on User interface software and technology*, ACM, 2003, pp. 61–70.

[137] Wobbrock, J. O., Rubinstein, J., Sawyer, M. W., and Duchowski, A. T. Longitudinal evaluation of discrete consecutive gaze gestures for text entry. In *ETRA '08: Proceedings of the 2008 symposium on Eye tracking research & applications*, ACM, 2008, pp. 11–18.

[138] Wu, M., Miller, R. C., and Garfinkel, S. L. Do security toolbars actually prevent phishing attacks? In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, CHI '06, ACM, 2006, pp. 601–610.

[139] Yee, K.-P. User interaction design for secure systems. In Deng, R., Bao, F., Zhou, J., and Qing, S., editors, *Information and Communications Security*, volume 2513 of *Lecture Notes in Computer Science*, pp. 278–290. Springer Berlin / Heidelberg, 2002.