Cognitive Authentication Schemes

Traditional password replacement?

M I C H A E L P A L M G R E N
a n d M A R K U S B Y S T R Ö M

KTH Computer Science
and Communication

Bachelor of Science Thesis
Stockholm, Sweden 2011

Cognitive Authentication Schemes

Traditional password replacement?

M I C H A E L P A L M G R E N
a n d M A R K U S B Y S T R Ö M

Bachelor's Thesis in Computer Science (15 ECTS credits)
at the School of Computer Science and Engineering
Royal Institute of Technology year 2011
Supervisor at CSC was Mads Dam
Examiner was Mads Dam

URL: www.csc.kth.se/utbildning/kandidatexjobb/datateknik/2011/palmgren_michael_OCH_bystrom_markus_K11008.pdf

Kungliga tekniska högskolan
Skolan för datavetenskap och kommunikation

KTH CSC
100 44 Stockholm

URL: www.kth.se/csc

Abstract

Although the traditional password authentication scheme is so widely used, it still has a few significant drawbacks. One of these weaknesses is that the strength of the password depends largely on the user. A poorly chosen password is weak against brute force attacks as well as dictionary attacks.

Due to this fact, there has been a lot of research regarding cognitive authentications schemes, where the user is authenticated with a system based on their answers to a number of cognitive challenges. This thesis conducts research on a few of these schemes and analyses their strengths and weaknesses. A cognitive scheme is also implemented and put through a usability test.

Our results suggest that the scheme implemented in this thesis is not practical as an authentication scheme for everyday use. The concept of cognitive authentication schemes does, however, show a lot of promise. A larger study would have to be conducted to come to a more sound conclusion.

Sammanfattning

Även om användningen av den traditionella lösenordsautentiseringen är så utbredd så lider den fortfarande av svagheter. En av dessa svagheter är att ett lösenords styrka beror mycket på användaren. Ett dåligt valt lösenord kan vara svagt mot “brute force” attacker samt “dictionary”-attacker.

Detta har lett till att man har forskat mycket inom kognitiva autentiseringsmetoder, där användaren autentiseras för ett system beroende på dennes svar på ett antal kognitiva utmaningar. I denna rapport studeras några av dessa metoder och deras styrkor och svagheter analyseras. En kognitiv autentiseringsmetod implementeras och sedan utvärderas användbarheten genom ett användbarhetstest.

Våra resultat från användartestet av metoden som implementerats i denna rapport är inte praktisk nog för vardagligt användande. Konceptet verkar däremot väldigt lovande. En större studie skulle behöva göras för att kunna komma till en rimlig slutsats.

Table of Contents

List of Figures

List of Tables

Section 1

Foreword

This document presents a bachelor's degree project in Computer Science, course DD143X, *Degree Project in Computer Science, first level* at School of Computer Science and Communication at the Royal Institute of Technology, KTH.

The purpose of this thesis is to evaluate if a Cognitive Authentication scheme can be implemented in such a way that it can replace or be used as a supplement to the traditional Password Authentication scheme that is used in nearly all IT-systems today to authenticate its users.

Markus developed the algorithm that calculates the correct passphrase in the application and did some general coding. Michael developed GUI for the application that implements the scheme, did some general coding as well as layout of the thesis.

Other than that, most of the work such as writing the report and the conduction of the user study was done together.

The supervisor and examiner for this thesis was Mads Dam.

Section 2

Document Overview

Section 3 This section introduces the reader to the weaknesses of the traditional password authentication, as well as the idea of cognitive authentication. Also contained is the purpose of this thesis, as well as its problem statement.

Section 4 This section describes a couple of existing solutions to cognitive authentication along with a list of their strengths and weaknesses. These properties will be taken into consideration for our own implementation of a cognitive authentication scheme.

Section 5 This section describes the method with which the cognitive authentication application was implemented. It describes our own requirements on the application - some derived from the analysis of the implementations described in the previous section, others proposed by ourselves. Other details regarding the implementation of the scheme is also contained here.

Section 6 This sections presents the details of the usability study conducted after the implementation of the cognitive authentication scheme. It contains details on how the study was conducted as well as the results of the study.

Section 7 This section contains the final discussion regarding the outcome of the implementation relating back to the problem statement formulated in Section 3.

Section 8 This section contains full details of what we have concluded as a result of the conducted research and usability study presented in Section 6.

Appendix This section contains the user survey form used in the usability study. This is in swedish.

Section 3

Introduction

3.1 Background

Definitions

- *Brute-force attack*: The idea of this attack is to systematically try all possible passwords until the correct one is found. A password's length determines how feasible it would be to perform a brute-force attack to obtain the password. The difficulty of finding a password with a brute-force attack grows exponentially with longer passwords.
- *Dictionary attack*: The idea of this attack is to determine a user's password by only trying likely possibilities in contrast to a brute-force attack. These possible passwords are usually derived from a list of words. Passwords made up of single words that can be found in dictionaries are very weak against this kind of attack.

Authentication Factors

- *Possession*: Something the user **has** in their posession. Includes ID cards and different types of security tokens (hardware tokens stored on a dedicated hardware device such as USB tokens or key fobs and software tokens stored on a general-purpose electronic device). Presentation of such a token does not, however, prove your ownership of the token or the identity connected to it, as the item could very well have been stolen or duplicated.
- *Knowledge*: Something the user **knows**. Includes passwords, passphrases, PIN numbers and challenge-response schemes.
- *Inherence*: Something the user **is** or **does**. Based upon the user's intrinsic traits, autonomal, physiological or behavioral. Includes fingerprint scanning, retinal scanning, signature checking and voice recognition, among other biometrical characteristics.

3.1.1 Password Authentication Scheme

Almost any computer system today that requires its users to authenticate themselves uses the traditional authentication scheme that has the user enter a secret of their choosing, i.e a password. Once a password has been entered, the system looks up the entered username and password in the password hash. If the system's stored password matches the entered password for the specified username, the user is authenticated with that system. This is called a *Password Authentication* (PA) scheme.

Weaknesses

Although this is the most deployed authentication mechanism, there is, however, a few significant drawbacks with this approach. For one, there is the human factor, and it's often said that - "the users are the weakest link in the security chain". In 2009, when `rockyou.com` was subject to an SQL injection attack which led to the public release of 32 million passwords, it was observed that about 30% of the users had chosen passwords with a length of six characters or shorter. Furthermore, about 60% used only a limited set of alphanumeric characters for their passwords [1]. Passwords of this sort are severely weak against bruteforce attacks and dictionary attacks.

Another problem presents itself when the user is logging in to a remote system. Since the password is sent over the network, the scheme becomes vulnerable to a misfeasor intercepting the traffic (eavesdropping) sent over the network. If this is the case, and the traffic is unencrypted (which is the case with many webservices today), then the eavesdropper has very a good chance of retrieving the unsuspecting user's password. This is a threat to both the integrity and confidentiality of the information on the user's account.

3.1.2 Cognitive Authentication Schemes

Cognitive Psychology is the part of psychology that explores the human internal mental processes. In other words, it explores our ability to acquire, process and use information. The four main areas are perception, memory, thought and linguistic processes [4].

By understanding these cognitive functions we can create systems that are easier and more intuitive to use.

An authentication scheme that utilises these cognitive functions is called a Cognitive Authentication (CA) scheme. These schemes might offer a solution to the aforementioned problems - either by replacing the PA scheme or using them as a supplement to the PA scheme. It has been shown in experiments that humans have an almost limitless memory for pictures [2]. And, according to the picture superiority effect, concepts are more likely to be remembered when presented as images rather than words [3]. Therefore, a possible candidate would be an authentication scheme that does not rely on a user's memory of a secret keyword, but instead on the cognitive

skills of the user. Krzysztof Golofit has conducted some research in this subject in his paper "Picture Passwords Superiority and Picture Passwords Dictionary Attacks" where he concluded that there was a *statistically* significant superiority of picture passwords over alphanumerical ones [5].

Cognitive passwords is an authentication concept that is based on a user's selected responses to one or more challenges posed by the system. There has been a lot of research done as well as several implementations of such a scheme. The results have been mixed, but this area of research shows a lot of promise as far as making passwords stronger, resulting in an increased difficulty of cracking them, but also easier to remember/recall. The problem is, however, how to implement such a scheme as it needs to be easily memorized, but at the same time hard to guess. Additionally, the authentication process should be relatively easy as to not disrupt the usability of the system itself.

3.2 Problem Statement

This thesis aims to answer the following questions:

- Can cognitive authentication schemes be used instead of the widely-used, traditional password authentication scheme?
- Is it better suited as a supplement to the traditional password authentication scheme to increase security?
- Is it practical?
- Is it secure?

Section 4

Theoretical Background and "State of the Art"

In this section, we describe three well-known cognitive authentication methods implementing graphical passwords as well as present what we consider their strengths and weaknesses. This analysis will then be taken into consideration during our implementation of a cognitive authentication scheme. This implementation is described in Section 5.

Definitions

- *Keylogger*: Hardware or software installed on a computer that tracks the keystrokes on a keyboard. As an attack, it is typically done without the knowledge of the person using the keyboard that their actions are being monitored.
- *Shoulder surfing*: A term that refers to a person using direct observation to gain information about, for example, another person's password. As the term suggests, this can be done by looking over the person's shoulder as they type in their password.
- *Multi-factor authentication*: A term that refers to a method of using two or more independent authentication schemes to grant a user access to a system. For example, using one *knowledge*-based method such as entering a password to get to the next method of authentication, which could be *inherence*-based. This would be called a two-factor authentication.
- *Offline attack*: A type of attack where an attacker has gained access to a password file and tries to crack these passwords on their own system by comparing generated password values with the hash values stored in the password file. As such, an attacker does not need to have any contact with an authorizing party.
- *Guessing attack*: A guessing attack is when an unauthorized user attempts to login to a system by guessing usernames and/or passwords. Brute-force attacks and dictionary attacks are the types of a guessing attack.

4.1 Weinshall's Cognitive Authentication

Procedure: During registration, a user is assigned a very large subset S from the set H that consists of all images in the scheme. They are then offered an extensive familiarization process where the user gets to memorize their images.

During authentication, a subset s of S are displayed among decoys in a matrix. The user then has to draw a virtual path between the images, starting from the top-left image. If the image the user is currently on exists in S, the user moves down. Otherwise, the user moves right. When the path has arrived at the label the bottom or to the far-right of the matrix, the user identifies the value at the label and enters that value in a multiple choice question. This process is repeated until the system determines that the possibility of a guessing attack has passed a certain threshold.

Figure 4.1: Concept design of a cognitive authentication scheme by Weinshall.

Information about this scheme can be found in [6], a report by D. Weinshall.

Strengths:
+ Safe against shoulder surfing, assuming that the user does not trace their path manually on the screen.
+ Safe against keyloggers.
+ Non-static authorization code.
+ No user choice regarding length of password.
+ Tolerates user error. This is good, since the user has to memorize a large number of images. Even by failing to remember one of the images in their set that was presented in the matrix or perhaps even unknowingly misclicking, the user can still be granted access and doesn't have to go through the trouble of going through the authentication process all over again.

Weaknesses:
- Requires an extensive familiarization period due to the large number of images needed to be recognized among the decoys in the matrix.
- Tolerates user error. This could be bad, since an attacker can theoretically gain access even by wrong guesses.
- Long login time.

4.2 PassPoints

Procedure: During registration, a user arbitrarily chooses n points from a chosen image.

During authentication, the user has to click on the chosen click-points (within a

certain tolerance radius). This has to be done either in the same order as when they registered or in any order, depending on the implementation.

Information about this scheme can be found in [7], a report by S. Wiedenbeck et al.

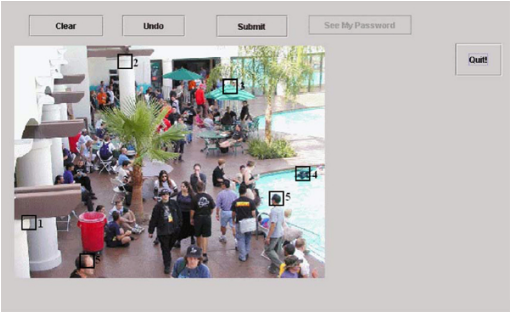


Figure 4.2: Concept design of PassPoints by S. Wiedenbeck et al.

Strengths:
+ Large password space due to the many click points in an image. This increases as the tolerance radius decreases.

Weaknesses:
- Static authorization code.
- Susceptible to offline attacks, as suggested by Julie Thorpe and P.C. van Oorschot in [9].

4.3 PassFaces

Procedure: During registration, a user is presented with a random set of faces (usually from 3 to 7) which will be their secret set. The user is then taken through a familiarization process.

During authentication, the user picks out their assigned faces one at a time from successive 3x3 grids. The user is authenticated when they have picked out faces from their secret set a number of times in a row. This number can be decided by the user.

Information about this scheme can be found in [8].

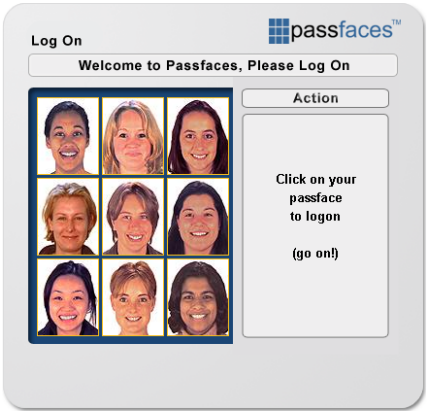


Figure 4.3: Design of PassFaces by Passfaces.com

Strengths:
+ Adds a non-time-consuming layer of security into a two-factor authentication environment.
+ Implicit mistake alert and server authentication if the user does not see their chosen images after entering their username and password. Two-way / Bidirectional authentication.

Weaknesses:
- Can supplement passwords, but could be weak on its own, due to user choice and its suceptibility to guessing attacks. If a user choses, for example, three rounds of challenges, their passfaces can be guessed in at most $9^3 = 792$ attempts. If the number of rounds is five, the number of attempts required would be at most

$9^5 = 59049$, which is a big improvement.
- In the case of eavesdropping and/or unencrypted traffic, an attacker can gain knowledge of the position of the images.

Section 5

Implementation Design and Method

Definitions

- *Password space*: The total number of password combinations.
- *Entropy*: When measuring the strength of an authentication protocol the term *entropy* is often used. Entropy is calculated by taking the base 2 logarithm (log_2) of the number of possible combinations of a passphrase. For example, in a combination lock with 4 digits in the sequence and where you can enter 0-9 per digit, you have 10^4 combinations. The entropy of that combination lock is therefore $log_2 10^4 \approx 13$. As such, a brute-force attack would require 2^{13} attempts to exhaust all possibilities. When the entropy value is increased by one, the attack is made twice as difficult.

Assumptions

- Although an increasing number of internet services encrypt the data between the clients and the server, we assume for the purpose of this thesis and scheme implementation that there is no encryption on the connection. From this assumption it follows that an eavesdropper knows all information sent between the client and the server: the images sent to the client, as well as their positions within the scheme and the code sent by the client to the server.

Personal Requirements

- The images presented in the application should be unique and distinguishable to make it easier to memorize them at registration as well as facilitate recognition among the other images during the authorization process.
- Because of the human factor, the number of images in a password should not be decided by the user and should instead be the same for all users. The number

of images in a password should be an appropriate tradeoff between password space and the number of images appropriate for memorization.

- Application must not show duplicates of the user's set of images nor the other generated images. This is to make the password unique and increase the password space, respectively.
- The passphrase is the challenge response, and is changed dynamically for each login session. The passphrase must not be directly connected to the user's set of images.
- The scheme should be practical and efficient. Login time (time to finish the challenges) should not exceed 60 seconds. It should not be perceived as a slow and tedious process.
- Login should be easy enough to not require any external assistance apart from eventual explanation of the authorization method.

The scheme we have implemented looks like this:

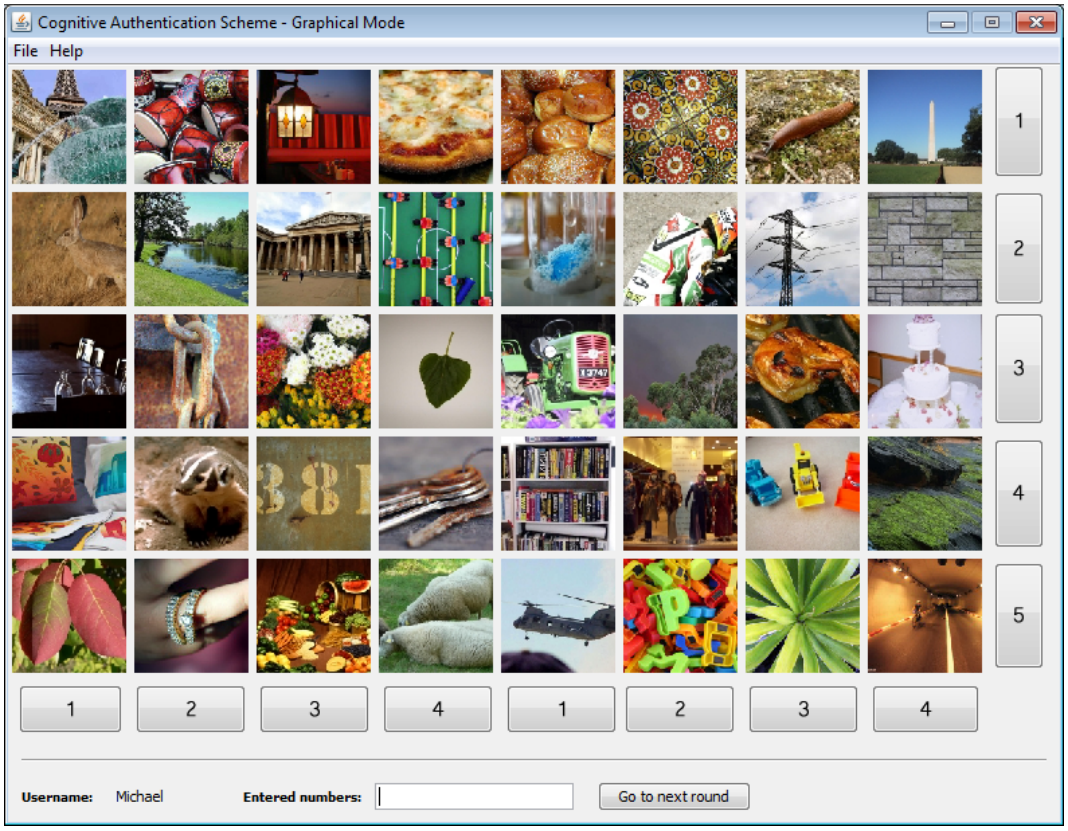


Figure 5.1: A screenshot of the implementation design of the Graphical mode of the authentication scheme.

Explanation of the authentication method: The authentication method that has been developed requires the user to create a virtual path in a matrix of images. On the bottom and right side of the matrix there are images which the user clicks when they exit the matrix. These buttons enter a number into a text area which will contain the session-unique passphrase when the user has completed the scheme. In the matrix there are 40 images of which 6 are in the user's set of secret images.

The process explained, step by step:

1. The user starts at the first unused image from their secret set which is as far up and to the left as possible in the matrix. This image is then "marked" as used.

2. When they have found the image, they go to the next picture to the right. Then the following rule applies for creating the path:

   - If the user is currently on one of their secret images they go down. Otherwise they go right.

3. The user repeats the process until they exit the matrix at the bottom or right side. At that point they click the button to which they exit the matrix. Then they keep going back to step 1 until all images are marked as "used".

When the user has created a path starting from each of the secret images, they click the "Proceed to next round" button, and are asked to repeat the process once more. If both of these processes are successful, the user is authenticated.

Motivation for the scheme design: A lot of inspiration for this particular authentication scheme has been taken from Weinshall's cognitive scheme in which you also create a path with the help of the secret images. The problem that we saw with Weinshall's scheme was that it took too long to successfully log in. Since only one path was created each round, the user had to do enough rounds so that the probability of the user guessing would pass a certain threshold. This meant that the user could have to do up to 11 rounds with a high complexity scheme and up to 22 rounds with a low complexity scheme.

This could possibly become frustrating for the user, so in our implementation the users create multiple paths each round in order to create a more complex passphrase that should be hard to guess. The implemented authentication scheme has a total of two rounds as a protection against an attacker sending random numbers to the scheme hoping that it is the passphrase. This increases the security while keeping the login times reasonably short.

The reason for the second step in our algorithm is due to a discovered flaw. If you immediately go down a row when you start at one of your "starting images" the path very often becomes the same, resulting in a passphrase with a very low variety of numbers.

The reason for having buttons numbered 1-5 on the right side, and buttons numbered 1-4 on the bottom was to make it more difficult for eavesdroppers to know which row or column the user exited the matrix. If the buttons were numbered 1-5 and 1-8, this would give an eavesdropper more information.

The decision to use a path in the authentication process was so that the secret images would not be directly linked to the passphrase that is sent to a possible server over an unencrypted network. If the process would just involve clicking on an image, an eavesdropper could identify the image that was clicked. The path also makes this scheme safe against shoulder surfing. The keyboard is not used at all in the authentication process, which makes the scheme resistant to keyloggers. In the event of a mouse logger being active when a user logs in, an attacker would know which buttons the user clicked. Knowing just this will not make the scheme vulnerable, but if the attacker knows which images were placed in what locations then they could reverse engineer the users images after multiple user logins.

We chose to set the number of images in a user's set of secret images to 6 because this seemed like an appropritate amount for memorization and password space. The motiviation for using a total of 40 images was on one hand the look of the GUI in the sense of making the scheme fit on a normal computer screen while keeping the images detailed/big enough and on the other hand for providing a sufficient password space.

The images diplayed each round are placed randomly in the matrix. This means that the passphrase that the user enters is non-static. So even if an eavesdropper captures the passphrase, it will be of no use.

In our implementation it is possible to enter numbers 1-5 and since 6 images from the user's set of secret images are displayed in the matrix at one time, the user will enter a total of 6 numbers. The number of possible combinations is therefore $5^6 = 15625$. This means that the probability of guessing the right passphrase when there is only one round is $\frac{1}{15625} = 0{,}000064 => 0{,}0064\%$. This assumes that the paths are evenly distributed which, however, is not the case for this implementation.

Guessing the right passphrase on an implementation with two rounds would then be $(\frac{1}{15625})^2 = 0{,}000000004096 => 0{,}0000004096\%$. This is a significant decrease of the success rate of an attack and the strenght of the two round scheme equals 28 bits of entropy.

The National Institute of Standards and Technology (NIST) has made an estimate of the bits of entropy in a user's password that consists of, for example, a word. The argument is that even if the password is made longer, the added characters do not increase the complexity of the password at the same rate as if the password would have been just random letters and numbers. This can be demonstrated with a simple example. Say a user has a word as the password. By knowing just a few of the characters in the password, we can greatly decrease the possible number of words that we must test in order to guess it. This is since the characters combined make some sort of meaningful word that might be in a dictionary or similar. If the user's password would have been just random letters and numbers, knowing a few of the characters doesn't limit what the other characters might be [10].

According to the NIST estimate, a "normal" 8 character user chosen password has an effective password entropy of roughly 21 bits. So, in that sense, the random passphrases (28 bits of entropy) that the user enter in our implementation is 128 times harder to guess. 28 bits of entropy would by NIST's estimate equal to a 15 character "normal" user password. Referring to NIST's estimates, an 8 character password with random mixed case letters as well as numbers has $log_2 62^8 = 48$ bits of entropy, but is at the same time difficult to remember.

To guess which secret images the user has would require at most 3.8 million tries in our implementation. That equals to 22 bits of entropy which makes it twice as difficult to guess as a "normal" 8 character password. More about this in the discussion.

Section 6

Usability Study

6.1 Method

The aim of this usability study is to study how a user interacts with the scheme we have implemented and their general attitude towards such a scheme.

The number of participants for this study was 15, 12 of which were men and 3 of which were women. The ages ranged from 15 to 62. Experience with computers were distributes as follows: "very little" (2), "good" (4), "very good" (4), "expert" (5)

The users registered their secret images to a database with one of two methods: they either chose their own images (7 participants) or were assigned images (8 participants) randomly from a collection of 240 images. This was done in order to find out if there was a significant difference in memorability of the images depending on the method used.

The rules for creating the path in the matrix were explained and the users had the chance to familiarize themselves with the scheme and practice a login-attempt with some assistance to make sure that they knew the algorithm before the test started. This did not take more than 5 minutes.

For the test, the users were asked to go through the login process five times. After each attempt, the time and whether the attempt was successful or unsuccessful was noted. Some of the participants (10) took the test again after three days. This was to see if the images would be remembered when some time had passed since they first came in contact with the scheme. The statistics in the next section were derived from the data collected during these tests.

Some users were also asked to test another variant of the scheme (henceforth called "Alphabetical mode"). This time, however, the images would be replaced by capital letters in different fonts . This was done to find out if using letters instead of images as a way of preserving some elements from traditional passwords would be more appropriate for this type of authentication scheme. Our implementation of such a scheme is shown in figure 6.1.

After all tests had been completed, the participants were asked to fill out a form

about their experience with the implemented scheme. This form is presented in the appendix in section 10.

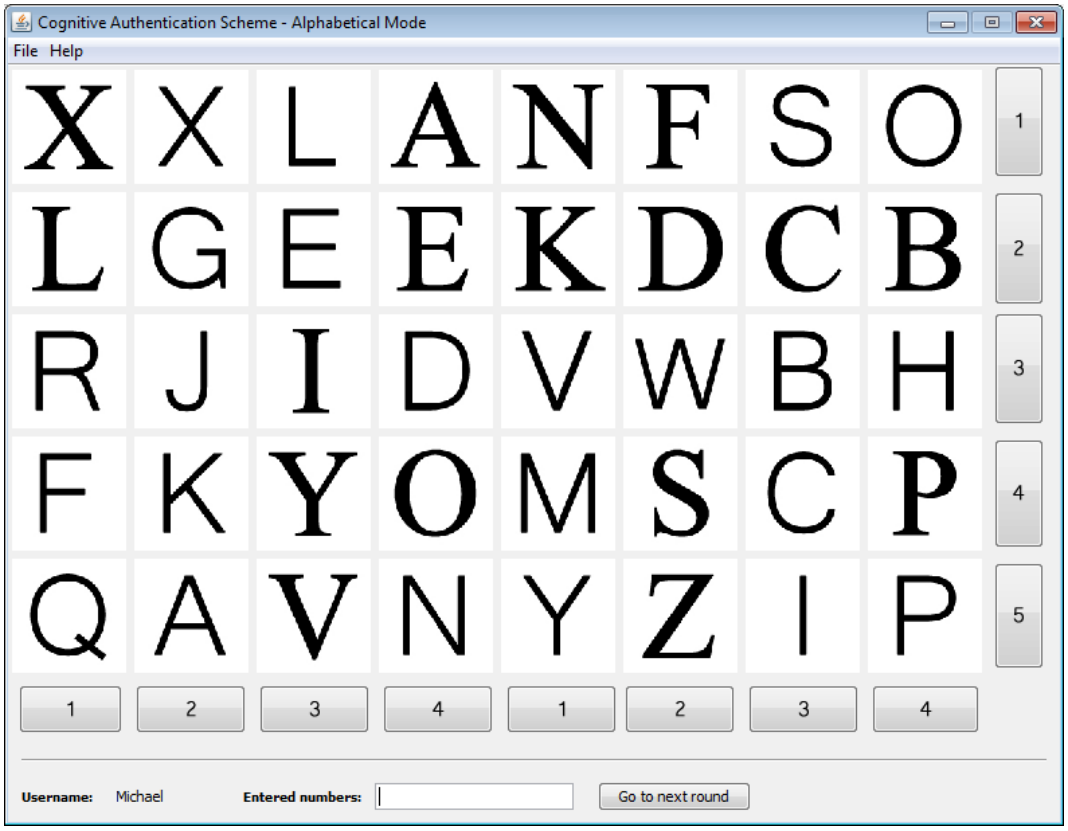


Figure 6.1: A screenshot of the implementation design of the Alphabetical mode of the authentication scheme.

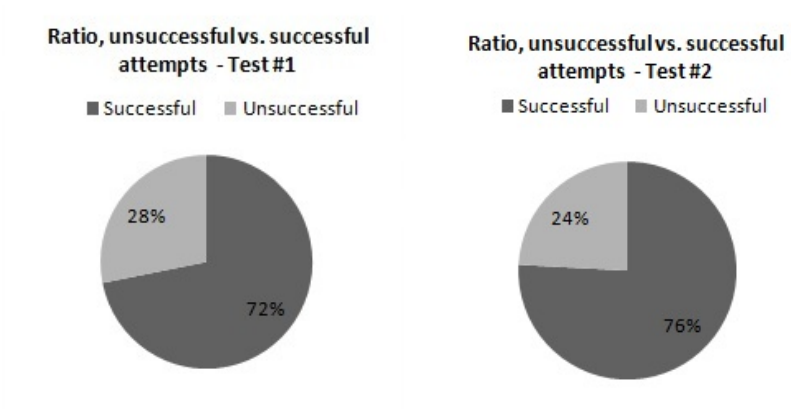
6.2 Results

The most surprising observation was how well the test subjects managed to remember their images. Also, there wasn't a significant difference between the users who had chosen their images and those that were assigned images. In some tests, the users who were assigned images did better than those who had chosen their own. All testers that took the test a second time after three days remembered their images.

Nearly everyone who took the test felt that the process of logging in was either quite long or long. 87% followed the path in the matrix with the mouse cursor. This is interesting since it allows a person looking over the user's shoulder to learn the user's secret images.

The diagrams below shows the ratio between successful and failed login attempts for

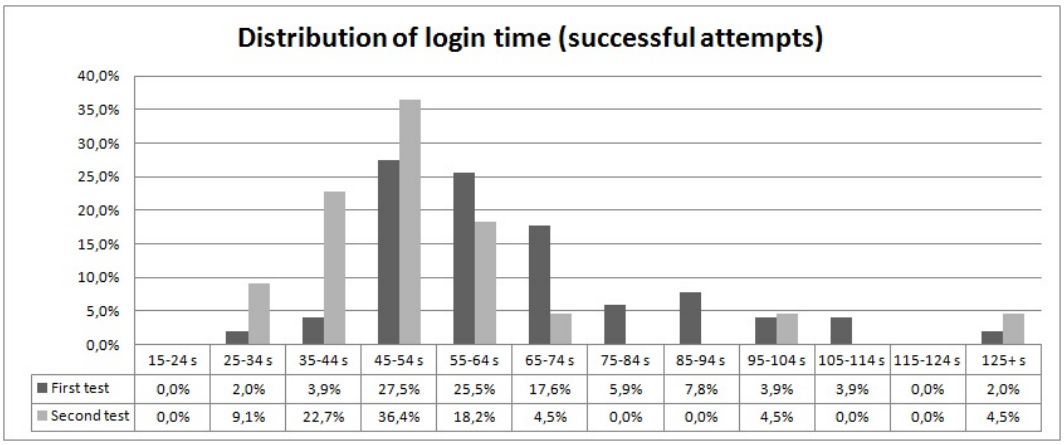
the first and second test, respectively. No *significant* difference was noted between the two tests.



(a) Success rate, test 1 (b) Success rate, test 2

Figure 6.2: Success rate, test 1 and test 2

The diagram below shows the distribution of login time for a successful login for the first and second test. No distinction was made between a successful login with chosen images and a successful login with assigned images. From this diagram, we gather that 58.8% of the successful login attempts were within 60 seconds (±4 seconds) on the first test, and 86.4% for the second test. In our opinion, this is a significant improvement.



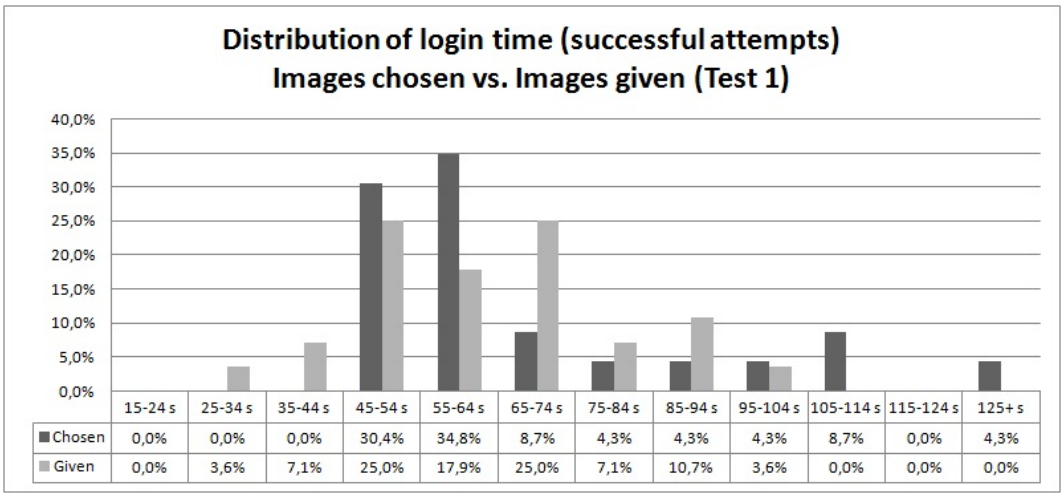
	15-24 s	25-34 s	35-44 s	45-54 s	55-64 s	65-74 s	75-84 s	85-94 s	95-104 s	105-114 s	115-124 s	125+ s
First test	0,0%	2,0%	3,9%	27,5%	25,5%	17,6%	5,9%	7,8%	3,9%	3,9%	0,0%	2,0%
Second test	0,0%	9,1%	22,7%	36,4%	18,2%	4,5%	0,0%	0,0%	4,5%	0,0%	0,0%	4,5%

Figure 6.3: Distribution of login times, first test and second test

Table 6.1: Slowest and Fastest logins, test 1 and test 2

	Test 1	Test 2
Slowest	188 seconds	237 seconds
Fastest	32 seconds	28 seconds

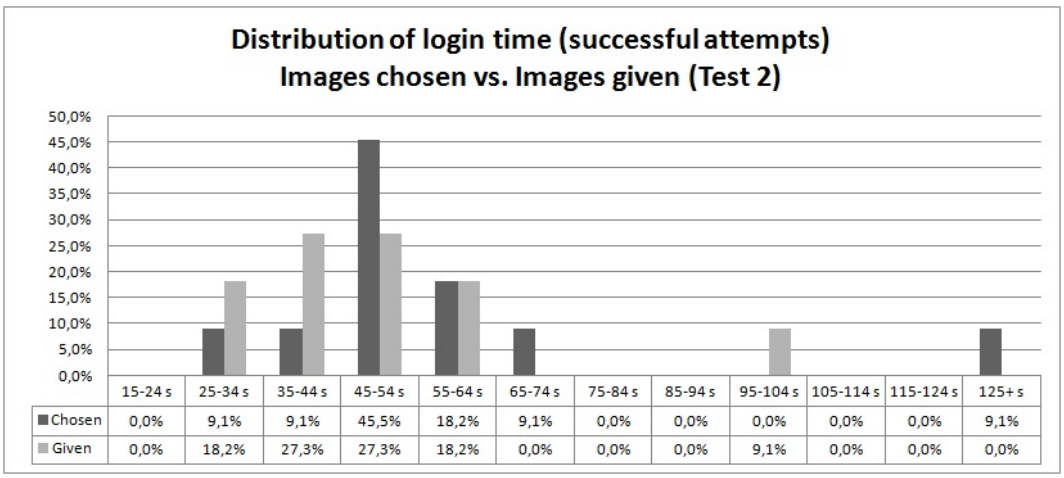
The diagram below shows the distribution of login time for a successful login for the first test, with a distinction between a successful login with chosen images and a successful login with assigned images.



	15-24 s	25-34 s	35-44 s	45-54 s	55-64 s	65-74 s	75-84 s	85-94 s	95-104 s	105-114 s	115-124 s	125+ s
Chosen	0,0%	0,0%	0,0%	30,4%	34,8%	8,7%	4,3%	4,3%	4,3%	8,7%	0,0%	4,3%
Given	0,0%	3,6%	7,1%	25,0%	17,9%	25,0%	7,1%	10,7%	3,6%	0,0%	0,0%	0,0%

Figure 6.4: Distribution of login times; Images Chosen vs. Images Assigned, test 1

The diagram below shows the distribution of login time for a successful login for the second test, with a distinction between a successful login with chosen images and a successful login with assigned images.



	15-24 s	25-34 s	35-44 s	45-54 s	55-64 s	65-74 s	75-84 s	85-94 s	95-104 s	105-114 s	115-124 s	125+ s
Chosen	0,0%	9,1%	9,1%	45,5%	18,2%	9,1%	0,0%	0,0%	0,0%	0,0%	0,0%	9,1%
Given	0,0%	18,2%	27,3%	27,3%	18,2%	0,0%	0,0%	0,0%	9,1%	0,0%	0,0%	0,0%

Figure 6.5: Distribution of login times; Images Chosen vs. Images Assigned, test 2

Table 6.2: *Experienced* difficulty to recall the secret images

Method / Difficulty	Easy	Quite Easy	Quite Difficult	Difficult
Chose Images	57%	29%	0%	14%
Assigned Images	63%	25%	12%	0%

Table 6.3: *Experienced* difficulty to recognize the secret images

Method / Difficulty	Easy	Quite Easy	Quite Difficult	Difficult
Chose Images	43%	43%	14%	0%
Assigned Images	50%	25%	25%	0%

The table below presents the results from the usability testing of the alphabetical mode of the implementation. It should be noted, however, that the success rate is very unrepresentative due to the limited sample size (only 3 people tested this variant of the implementation) and all participants thought it was so difficult, time-consuming and generally boring that they wanted to give up.

Table 6.4: Average login time and success rate, Alphabetical mode

Average login time (s)	126
Fastest login time (s)	59
Slowest login time (s)	221
Success rate (%)	73

Section 7

Discussion

We propose that our implementation of a cognitive authentication scheme is safe against keyloggers and safer against eavesdropping compared to traditional passwords. Since the authentication keys are non-static, it also makes brute-force attacks much more difficult and dictionary attacks virtually impossible.

Furthermore, we propose that it is much stronger against shoulder-surfing compared to traditional passwords, given that the user does not follow their path through the matrix, essentially giving away the position of their images.

A similar weakness exists in our implementation. The position of the images could quite possibly be derived from the sequence of numbers in the passphrases entered by the user after a path through the matrix has been completed. This could possibly be done by monitoring multiple successful login attempts.

In our study, we did not notice any significant difference in login success rate between the users who were assigned images and those who chose images themselves. However, some users who chose images did so according to a certain theme or with certain criterion thinking that this would make it easier to recognize the images in the matrix. Since several images appearing in the matrix - regardless if the image belongs to the user's secret set of images - could follow the same criterion, this makes it difficult to distinguish the images. This was not a problem for the users who were assigned passwords, as these images did not follow any criteria.

The feedback received from the subjects that tested the alphabetical mode made it clear that such a scheme was heavily inferior to the image-based version. This feedback, along with the fact that testing this scheme in the same extent as the image-based version would double our testing time, we decided to not test this scheme as extensively.

The algorithm that the path-making was based on felt complicated according to the users. If encryption would be used over unsecure connections, this process could be made much easier. However, logging in by clicking on the images makes the scheme vulnerable to shoulder surfing.

There is also a another potential problem with these kinds of cognitive authentication

schemes. If two systems use the same scheme with the same images, and a user has a different set of secret images on these two systems, then they might have a problem differentiating the two systems images from eachother. This observation was made in our study.

Section 8

Conclusion

The concept of cognitive passwords looks promising. Remembering images seemed easy for all test subjects, which would suggest that a similar scheme would be possible.

Based on the responses received in the study, such a scheme would have to be made simpler and be less time consuming than the one implemented for this thesis.

One thing is for sure, using images instead of alphabetical letters as stimuli for the login process in a cognitive authentication scheme such as this one is far superior in the sense of usability.

However, a more elaborate study would have to be conducted - with a larger number of test subjects with a wider variety of experience with computers and over a longer period of time - to come to a more sound conclusion.

Our own implementation of a cognitive authentication scheme is not as secure as a randomly generated password with, for example, 8 characters with mixed case and numbers. It is, however, generally more secure than a 8-character user-chosen password.

Used as a supplement to traditional passwords, another layer of security is added. It also makes the authentication a two-way authentication. We do not believe that our own implementation would be a good candidate for a supplement as it is initially designed to be a standalone authentication scheme. We believe, however, that a simpler scheme can be developed to be used as a supplement, such as Passfaces, but a little more secure.

If you ignore the threat of eavesdropping and instead focus on the other security threats mentioned in this thesis (guessing attacks and shoulder-surfing), we believe that a scheme that is simpler for the user, more efficient in terms of login time and more secure against the aforementioned attacks can be developed, as, in our own experience, the threat of eavesdropping put some limits on our own implementation.

References

[1] The Imperva Application Defense Center (ADC), 2010. *Consumer Password Worst Practices*,
Available at: http://www.imperva.com/docs/WP_Consumer_Password_Worst_Practices.pdf
Date accessed: 2011-02-27

[2] Standing, L.; Conezio, J.; Haber, R. N., 1970. *Perception and memory of pictures: Single-trial learning of 2500 visual stimuli.* University of Rochester. Rochester, NY
Available at: http://cvcl.mit.edu/SUNSeminar/standing70.pdf
Date accessed: 2011-03-01

[3] Wikipedia.org.
http://en.wikipedia.org/wiki/Picture_superiority_effect

[4] Karlsson, M., 2006. *Lecture slides in course "HIIA20 Användbarhet med kognitiv psykologi"*. Linköpings universitet,
Available at: http://webstaff.itn.liu.se/~marka/TNMK31-2006/fo3.pdf
Date accessed: 2011-04-10

[5] Golofit, K., 2007. *Picture Passwords Superiority and Picture Passwords Dictionary Attacks*, in Proceedings of the International Multiconference on Computer Science and Information Technology, pp. 681 690
Available at: http://www.proceedings2007.imcsit.org/pliks/139.pdf
Date accessed: 2011-03-01

[6] Weinshall, D., 2006. *Cognitive Authentication Schemes Safe Against Spyware (Short Paper)* in Proceedings IEEE Symposium on Security and Privacy (2006)
Available at: http://www.cs.huji.ac.il/~daphna/papers/Weinshall_S%26P_2006.pdf
Date accessed: 2011-04-14

[7] Wiedenbeck, S. et al., 2005. *PassPoints: Design and longitudinal evaluation of a graphical password system* in Int. J. Human-Computer Studies 63 (2005) pp. 102 127
Available at: http://clam.rutgers.edu/~birget/grPssw/susan1.pdf
Date accessed: 2011-04-10

[8] passfaces.com, *Passfaces Technology*,
http://www.passfaces.com/enterprise/about/about_passfaces.htm
Date accessed: 2011-04-12

[9] Thorpe, J; van Oorschot, P. C., 2007. *Human-Seeded Attacks and Exploiting Hot-Spots in Graphical Passwords*. School of Computer Science, Carleton University
Available at: `http://www.ccsl.carleton.ca/paper-archive/usenix07.hotspots.pdf`
Date accessed: 2011-04-10

[10] Burr, B., 2004. *NIST E-Authentication Guidance SP 800-63* (2008). National Institute of Standards and Technology (NIST)
Available at: `http://csrc.nist.gov/archive/pki-twg/y2004/Presentations/twg-04-04.pdf`
Date accessed: 2011-04-05

Appendix

ANVÄNDBARHETSSTUDIE

Kognitiva autentiseringsmetoder

Kön…

☐ Man
☐ Kvinna

Ålder…

☐ 15 – 20 år
☐ 21 – 26 år
☐ 27 – 32 år
☐ 33 – 38 år
☐ 39 – 44 år
☐ 45 – 50 år
☐ 51 – 56 år
☐ 57 – 62 år
☐ 63 – 68 år

Datorvana…

☐ Ingen
☐ Mycket liten
☐ God
☐ Väldigt god
☐ Expert

Om du fick välja bilder själv, gå till del 1. Annars, gå till del 2.

Del 1

Hade du egna kriterier när du valde ***vanliga bilder***?

☐ Ja ☐ Nej

Om ja, vilka var kriterierna? ……

Kryssa i det alternativ som bäst beskriver hur lätt/svårt det var att komma ihåg de **vanliga bilderna** du valde?

Precis efter val av bilder…

☐ Lätt ☐ Ganska lätt ☐ Ganska svårt ☐ Svårt

Efter 3 dagar…

☐ Lätt ☐ Ganska lätt ☐ Ganska svårt ☐ Svårt

Kryssa i det alternativ som bäst beskriver hur lätt/svårt det var att känna igen dina bilder i matrisen.

☐ Lätt ☐ Ganska lätt ☐ Ganska svårt ☐ Svårt

Hade du egna kriterier när du valde ***bokstavsbilder***?

☐ Ja ☐ Nej

Om ja, vilka var kriterierna? ……

Kryssa i det alternativ som bäst beskriver hur lätt/svårt det var att komma ihåg **bokstavsbilderna** du valde?

Precis efter val av bilder…

☐ Lätt ☐ Ganska lätt ☐ Ganska svårt ☐ Svårt

Efter 3 dagar…

☐ Lätt ☐ Ganska lätt ☐ Ganska svårt ☐ Svårt

Kryssa i det alternativ som bäst beskriver hur lätt/svårt det var att känna igen dina bilder i matrisen.

☐ Lätt ☐ Ganska lätt ☐ Ganska svårt ☐ Svårt

*Fortsätt till **del 3**.*

Del 2

Kryssa i det alternativ som bäst beskriver hur lätt/svårt det var att komma ihåg de **vanliga bilderna** som du blev tilldelad.

Precis efter bilder blev givna…

☐ Lätt ☐ Ganska lätt ☐ Ganska svårt ☐ Svårt

Efter 3 dagar…

☐ Lätt ☐ Ganska lätt ☐ Ganska svårt ☐ Svårt

Gjorde du något speciellt för att komma ihåg bilderna du blev tilldelad?

☐ Ja ☐ Nej

Om ja, vad gjorde du?: ……

Kryssa i det alternativ som bäst beskriver hur lätt/svårt att känna igen dina bilder i matrisen.

☐ Lätt ☐ Ganska lätt ☐ Ganska svårt ☐ Svårt

Kryssa i det alternativ som bäst beskriver din uppfattning av att komma ihåg **bokstavsbilderna** som du blev tilldelad.

Precis efter bilderna tilldelades dig…

☐ Lätt ☐ Ganska lätt ☐ Ganska svårt ☐ Svårt

Efter 3 dagar…

☐ Lätt ☐ Ganska lätt ☐ Ganska svårt ☐ Svårt

Gjorde du något speciellt för att komma ihåg bilderna du blev tilldelad?

☐ Ja ☐ Nej

Om ja, vad gjorde du?: …….

Kryssa i det alternativ som bäst beskriver din uppfattning av att känna igen dina bilder i matrisen.

☐ Lätt ☐ Ganska lätt ☐ Ganska svårt ☐ Svårt

*Fortsätt till **del 3**.*

Del 3

Uppfattade du inloggningsprocessen som krånglig (sättet att gå genom matrisen)?

☐ Ja ☐ Nej

Om ja, varför?: …….

Hur lång tid tog det för dig att förstå inloggningsprocessen efter en förklaring och demonstration?

☐ Inom några minuter ☐ Inom en halvtimme

☐ Inom en timme ☐ Förstod aldrig helt

Om du tänker tillbaka på när du loggade in, följde du då vägen genom matrisen med muspekaren?

☐ Ja, ofta ☐ Ja, ibland ☐ Nej

Vad upplevde du som svårast med registrering/inloggning?

Svar: …….

*Fortsätt till **del 4**.*

Del 4

Om denna inloggningsmetod var minst lika säkert som ett vanligt lösenord, skulle du kunna tänka dig att använda dig av denna typ av inloggning istället för ett lösenord? Varför/Varför inte?

Svar: ……

Vad tyckte du om tiden det tog att logga in, upplevdes den som lång/kort?

☐ Kort tid ☐ Ganska kort tid ☐ Ganska lång tid ☐ Lång tid