PASSWORD CUEING WITH CUE(INK)BLOTS

Karen Renaud, Anthony McBryan, Paul Siebert
Department of Computing Science, University of Glasgow
18 Lilybank Gardens, Glasgow, G12 8RZ. United Kingdom
{karen,mcbryan,psiebert}@dcs.gla.ac.uk

ABSTRACT

People forget passwords daily, and this leads to frustration and potential loss of revenue commercially. Mechanisms for proving identity in the face of forgotten passwords are mostly unsatisfactory, because they are so insecure. The problem is that it is difficult to handle password replacements efficiently. One of two people could be requesting the replacement: the legitimate user or a potential intruder. Unfortunately, the system doesn't have any way of knowing which it is. Some systems make an effort to confirm identity by using one or more challenge questions. The user provides the answers to these questions at enrolment, and the rationale is that if the requestor can provide the same answer later, it must be the same user. Of course this assumption is flawed because the answers could be discovered or known by an intruder. An alternative to challenge questions, explored in this paper, is the use of an abstract image to elicit a textual description. This description could be used as the password, or as an alternative to the challenge questions. We report on an experiment which tested images as cues.

KEYWORDS

Authentication, cueing, visual perception, security.

1. INTRODUCTION

The 21st century user makes use of a number of secure sites which require confirmed identification of users. Since proof of this identity is most often achieved by means of a secret password, the direct consequence is that people have a number of passwords to remember.

Since human memory is fallible, this leads to regular forgetting episodes, which, in turn, requires a password replacement mechanism. When people forget things in a less secure context, one can provide them with a cue, but cueing in an authentication setting is challenging. One has to deal with the possibility that the person requesting the cue could well be an intruder and the cue, in this case, could be well aid the intruder. What is needed, therefore, is a cue that makes sense only to the legitimate user. This is the rationale behind challenge questions that tap into our childhood memories, which tend not to fade as easily as recent memories. Unfortunately the answers to these questions are seldom secret and are relatively easy to discover.

Users attempt to prevent forgetting episodes by choosing weak passwords or by using the same password for all accounts [6]. Password strength is traditionally measured in terms of entropy. This is the measure of uncertainty an attacker has in guessing the password. If we assign a user a random password using alphabetic characters (there are 94 characters available), then the entropy of password will be 6.55 bits per character. However, random passwords are easily forgotten, so people usually use a word in their language of choice. In a language such as English that reduces the entropy to 2.3 bits per character. This is low because people usually only make use of lower case characters and words they can remember. To alleviate the problems caused by weak passwords we should rather find ways of helping users to remember their passwords.

We have shown, in a previous study [1], that a specific kind of abstract image has some potential as a cueing mechanism. The research presented in this paper was undertaken to determine whether the presence of a cue "in the wild" would give users sufficient confidence in the cueing ability of the image description that they would choose to use stronger passwords.

In Section 2 secret-based authentication is reviewed. In Section 3 we briefly review the literature on forgetting and cueing. Section 4 gives information about the methodology followed in order to test the use of cueblots in authentication. Section 5 presents the results of the experiment and Section 6 concludes.

2. AUTHENTICATION

In order to grant access to a restricted digital space, we use a two phase protocol: identification followed by authentication. Users are identified by means of a text string — either an email address or a special user name — and then authenticated to verify the identity. Authentication can be verified by means of a shared secret, called a password or pass-phrase, or by means of a biometric which measures the user's physiology or behaviour and matches it to a previously recorded template. Users can also be authenticated by means of a token such as a smart card, but this is usually paired with a PIN, which the user has to remember, which has the same memorability problems as passwords. Since biometric measuring devices are far more expensive than keyboards, and cannot be trusted if deployed in an untrusted environment or over an untrusted network, most authentication these days is done by means of a secret password.

There is nothing inherently wrong with this mechanism — it is well established and socially acceptable. Passwords have been used for centuries, for example, by Roman centurions [11] and in the use of "Open Sesame" in the well known tale of Ali Baba. This makes them a well-understood and accepted paradigm.

People easily remember one password; problems start when they have to remember many passwords. Human memory is fallible. People will forget their passwords and these have to be replaced so that users can access their electronic accounts.

One of the most common mechanisms, used by a variety of websites, is the use of challenge questions to prove identity when users forget their passwords. On the face of it this is a viable mechanism but a closer look at challenge questions reveals many flaws. One has two choices in posing questions —either the user chooses his or her own questions at enrolment, and provides the answers, or the system provides a set of questions, and the user chooses to provide the answer to one. Both options have problems:

- If the user has to generate the question he or she is equally likely to forget the question as the password. In this case the cue question places an extra demand on the user's memory and is equally vulnerable to decay. One also has to put software in place to ensure that users specify a reasonable and well-formed questions and do not simply enter their own password as the question, for example.
- If the system provides a set list of questions, these need to be applicable to a wide range of users. The site developers usually resort to setting widely applicable questions based on the person's past, requesting details such as the name of the person's first school, first pet or mother's maiden name. The fatal flaw with these questions is that a relatively superficial knowledge of the legitimate user is required in order to know the answers to these questions, and the challenge questions thus offer an intruder a convenient and insecure way into the system. Even if the answer to the chosen question is not easily determined using research, the fact that most systems revert to the same set of questions makes a mockery of the supposed "secrecy" of the answer. The more sites holding the answers, the less of a secret they are.
- The answers to these questions are easily ascertained by a determined intruder, using social engineering [13].

Some systems prefer not to make use of challenge questions and revert to emailing forgotten passwords to users. This is also an insecure practise because email is seldom encrypted and is easily intercepted by a hacker. The use of one-time passwords, which require changing as soon as the person logs in, is somewhat more secure, but only if it is indeed the legitimate user who is trying to gain access. If an intruder is requesting the password reminder, and watching for the email, the legitimate user will probably be completely unaware of the intrusion into his account until the negative effects of the intrusion manifest themselves. In conclusion, the current replacement processes are unsatisfactory, because:

1. A key replacement process weakens the mechanism because the key has to be delivered in some way and this delivery is sometimes intercepted by an intruder who then proceeds to impersonate the legitimate user.
2. An alternative identity verification mechanism, such as challenge questions, weakens the mechanism because the answers are far more predictable than a secret password.

3. The replacement has to be funded and the cost is anything but negligible. Gartner [14] claims that a single replacement costs between $15 and $30. They estimate that each employee will call about 5 times a year (since they have passwords for multiple systems).

We have to address the inherently unmemorable nature of passwords because the mechanism is weakened when people write down their passwords in order to remember them or ask for replacements which are intercepted by intruders. Since forgetting causes many of the problems related to passwords, the next section takes a closer look at this human propensity.

3. FORGETTING

Humans learn in two ways — explicitly and implicitly. Implicitly learnt skills seldom decay but explicitly learnt knowledge is extremely prone to decay. Unfortunately humans do forget their passwords, and the forgetting is seldom deliberate. Most forgetting occurs early on in the process and then slows down later on [4]. Thus details may be forgotten within minutes if no serious attempt is made to encode the information in more than a cursory fashion.

Consider, now, how most passwords are chosen. Someone visits a website and is asked to provide a password, which is to be used at future visits for authentication purposes. The person's goal is to peruse the website, or purchase some items, or perhaps something else – but whatever it is, the definition of the password is probably extraneous to the person's immediate goals and purposes. If the person has had experiences of forgotten passwords in the past s/he may have a well-worn password that s/he uses for this kind of eventuality, and s/he provides it. If s/he is concerned about security and is wary of using a previously-used password, s/he may provide a unique password and write it down. If, however, s/he chooses to provide a password but not to record it, s/he will be allowed into the website and the password is likely to disappear into the mists of time, especially if s/he uses the site infrequently.

Schacter [10] calls this the sin of transience. Schacter cites a number of memory improvement programs and health cures and concludes that none are miracle cures. One thing that does assist effective retrieval of remembered facts is the effectiveness of the encoding process. Schacter cites research into a mechanism called elaborative encoding where the person spends some time encoding the information using visual imagery, mnemonics or elaborative questions. These are indeed effective but, of course, require extra effort from the person and are unlikely to be used in an uncontrolled password defining setting.

Another way of improving retrieval is by the provision of cues. Nyberg et al. [9] argues that retrieval of information activates the same brain regions as those activated when the information was encoded. They experimented with word-sound encoding and found that provision of the sounds assisted word retrieval. Moscovitch and Craik [8] found that cueing was beneficial at deeper levels of encoding.

A cue can be defined as: a. *A reminder or prompting*, or b. *A hint or suggestion.* A cue heard by a potential intruder, therefore, could help the intruder as much as the legitimate user, especially if the cue is easily understood. In an authentication setting such a universally accessible cue is useless since it undermines the strength of the authentication key.

Instead of resorting to password replacement, we should try to find a cue that makes sense to the legitimate user but not to an intruder. Hertzum [7] proposes that users specify particular password characters which will be displayed at password entry in order to jog their memory. This idea was tested with 14users and it was found that it did help them to remember their passwords. Hertzum notes that the defined passwords were often weak and predictable and argues that some kind of cueing mechanism is required in order to support the use of longer and stronger passwords. The implication is that users, if given a reliable cueing mechanism, will behave more securely and choose stronger passwords.

The possibility we have explored is the use of special images. The idea is that the user is given a personal image at enrolment and is asked to provide an image description of the image. This description can be used either as the password itself or, alternatively, as a personalised and less guessable challenge question. Of course a purely representational image will not work in this secure context because what one really needs is an image that generates a different textual association for different users so that intruders cannot confidently guess textual associations within the three strikes allowed before a lockout.

Stubblefield and Simon [12] experimented with abstract images they called inkblots which were intended to assist users in forming a semantic association with the textual password, which could be used as a reminder mechanism as required. They displayed 10 inkblot-like images in a particular sequence.

We call them inkblot-like because the term "inkblot" is most commonly associated with the controversial Rorschach inkblots, which were composed of a fixed set of 10 outlined images. These inkblots were used by psychiatrists to diagnose personality disorders but this practise has now largely been discredited [15].

The kinds of images used by Stubblefield and Simon were coloured and generated by a computer and thus do not really fit in with traditional notion of an inkblot. At each authentication attempt the "inkblot" was displayed and the user in Stubblefield and Simon's experiment was required to enter two characters — the starting and ending character of their inkblot description. They had some success in trials of this mechanism, achieving an entropy of 4.09 bits per character. However, the cognitive load imposed on the user is significant. They do not merely provide a textual description; they have to parse it in their minds to extract the required starting and ending character, and then type that in. Stubblefield and Simon do not give demographic information about their experimental subjects but one can envisage this cognitive load being untenable for any but the most mentally agile of users.

Our hypothesis is that we could make use of images as cues, but the following needs to be investigated first:

1. What kinds of images would elicit a textual description that the user could remember and that would be strong enough to be used as a password?
2. How durable are the image text associations?
3. Is it possible to necessitate the right level of cognitive processing so that the user remembers the password but does not feel overloaded? The mere provision of an image at key encoding time will not facilitate error-free retrieval later unless the person engages in some cognitive activity related to the image. Re-processing an image to re-generate a textual description at retrieval time should, in theory, excite the same parts of the brain as those involved in formulating the original textual description, and deliver the same textual description in a shorter time period on each access until the retrieval becomes automatic and the cue is no longer required.

We conducted a series of experiments to answer these questions. The first and second questions were investigated by means of an experiment which compared different abstract image types in terms of convergence of image descriptions and memorability of those descriptions. The results of this experiment are reported in [1]. It was clear that inkblot-type images had the most potential for cueing purposes. We have called these specially produced inkblots *cueblots*.

The third question was explored by means of a follow-up experiment which tested the use of cueblots in authentication over an extended period. The results of this experiment are reported here. There are two possible uses of cueblots — either to use the cueblot textual description as a password, or to use it in place of challenge questions when the password is forgotten. For the purposes of this study we decided to use the cueblot description as the password. This gives us more data about the use of the cueblot than if we used it only when users forgot their passwords.

4. CUEBLOT CUEING

In order to test the efficacy of cueblots we developed a website for an elective module at the University of X. The website gave students access to lecture notes, their grades and various other resources. A total of 53 undergraduate students used the website. Users were randomly assigned to receive a password with or without a cueblot. The cueblot-assisted authentication process had the following phases:

1. *Registration*: users were given a user name and registration code, by email, to facilitate the registration process. When they entered the key, the system either allowed them to choose a password (for the password condition), or displayed a cueblot, and allowed the user to customise and tailor the cueblot, as illustrated in Figure 2, to his or her satisfaction. The user was then instructed to give a cueblot description as a password.

2. *Authentication*: The user entered his or her user name and was directed to the authentication page. In the case of password users a simple password text entry area was supplied. In the case of cueblot users the cueblot was displayed and the user could re-enter the cueblot description, as shown in Figure 3.

3. *Replacement*: users could request a re-registration from the website administrator by email if the password had been forgotten.

The experiment ran for 9 weeks and all accesses were logged to facilitate analysis. The results are presented in the following section.

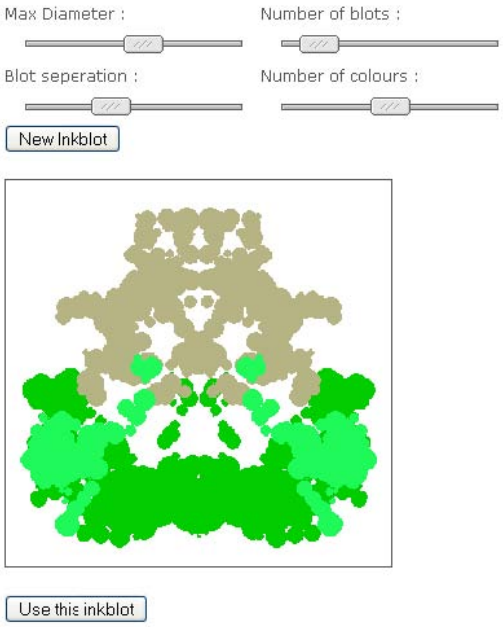

Figure 2: Enrolling with a Cueblot

5. RESULTS

Of the users who had agreed to their login behaviour being monitored a total of 53 actually used the site. Of these, 24 were allocated to the password condition and 29 to the cueblot condition. One user from the password condition needed a password reset during the course of the experiment and both the original and replacement passwords are included in our analysis; no users in the cueblot condition requested a replacement password.

We encountered six instances of people who deviated from the instructions provided for their condition. Two password users used the registration code as their password, probably because they had an email record of this code, and this made things easier for them if they forgot their password. Four people chose to ignore their cueblot, declining the offered cue and instead providing a password or pass-phrase of their own choosing. Since this type of behaviour is entirely possible in real life deployments, we retained these passwords/descriptions throughout our analysis.

Authentication mechanisms, whether they make use of cues or not, must try to maximise both ease of use and security — with neither taking the upper hand. The next two sections will consider our findings of cueblots-

assisted authentication in terms of these perspectives. The main aim of this experiment was to determine whether the level of cognitive processing required in using cueblots to cue passwords was acceptable to users. In addition to the quantitative analysis of logging records, we analysed responses to a questionnaire we asked all website users to complete.

5.1 Security

When discussing the security of an authentication scheme based on textual input the first measure considered is typically the length of the password and its character complexity. That is to say; longer passwords with larger choices of available characters (i.e. lowercase and uppercase letters, numbers and special characters instead of just lowercase letters) will result in much more secure passwords.

When we consider the length of the response for each condition we find, surprisingly, that there is no significant difference between passwords (M: 7.52, SE: 0.332) and cueblots (M: 8.31, SE: 0.632), $p>0.05$. Similarly when we evaluate the mean number of bits required to encode a character within a response for passwords (M: 2.49, SE: 0.09) and cueblots (M: 2.64, SE: 0.11)we find that this, too, is not significant, $p>0.05$.

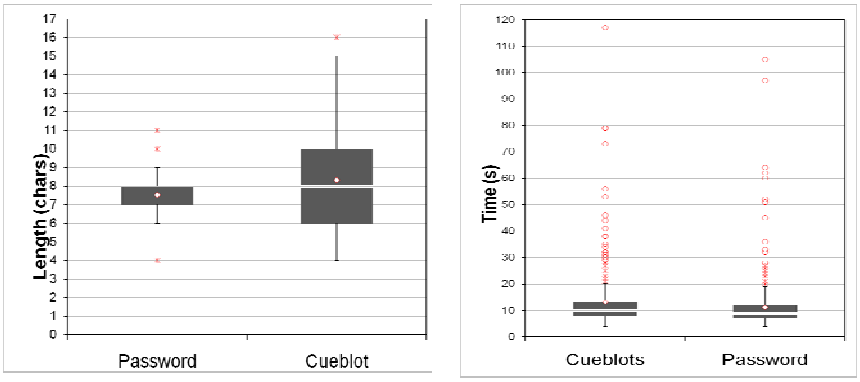

Figure 4: Mean Length of Response & Login Time

The length and number of bits per character, however, do not tell the full story. We also have to consider how similar descriptions are to each other and to what extent they have similar substrings. We first consider the Levenshtein distance between strings (otherwise known as the edit distance). This measures how many transformations are necessary to convert one string into another and therefore a higher score is desirable. Since this is highly correlated with the length of the responses involved we normalise the value by the length of the string after applying a heuristic method to compare the string to every other string. We find that the difference between Levenshtein scores calculated for passwords (M: 0.53, SE: 0.044) and cueblots (M: 0.6, SE: 0.033)is not significant, $p>0.05$.

We repeated this exercise using the Smith-Waterman algorithm, which is designed to do local sequence alignment. This allows us to measure the longest common sequences between strings (i.e. common uses of words such as " the "), in this case a higher score indicates a longer sequence and thus a lower score is desirable. We found that cueblots (M: 0.08, SE: 0.005) had a significantly higher Smith-Waterman score than passwords (M: 0.05, SE: 0.006), $t(48.85) = -4.088$, $p<0.05$, which indicates that users often include a subset of common words within their cueblot descriptions.

These results appear to conflict with the results in our previous work [1], where we identified cueblots as the image with the most potential as an authentication cue because when different users were shown the same cueblots they gave longer and more varying answers than for any other type of abstract image. Our result confirms findings by Brostoff et al.[3]during evaluation of the Passfaces authentication mechanism.

In this experiment it seems that when the user knows that the cueblot description is going to be used frequently as the password, he or she provides a much shorter description than would be provided if the

description was only going to be provided once or twice. This is perfectly reasonable, because users emphasise convenience over security. Hence the length of response and bits per character are basically the same as passwords. This is rather disappointing since we had hoped that the presence of the cueblot would allay users' fears of forgetting their passwords and therefore encourage them to choose stronger passwords.
When we first began our research into this area we believed that users would embrace the ability to create longer passwords if they were provided with a way to help them remember the password more easily. Unfortunately this does not appear to be the case. Our results indicate that the descriptions offered by users for the purposes of passwords are of comparable length and complexity to traditional passwords but with the problem that users will tend to include common stop-words in their description, which weakens the password. In the next section we will analyse the users' performance at using cueblot-assisted authentication in the context of time and effort required as well as login success rates

5.2 Ease of Use

In this section we focus on the results gathered from our experiment which give us an indication of the user's experience of using the cueblot system as compared to the traditional password system.

We start at the logical beginning and discuss our *registration* procedure. Users were sent a registration code by email, which allowed them to choose a password or to design a cueblot and then enter its description. Although it is often glossed over, the registration of a system can play a vital role in forming the user's initial perspective of the system. In our experimental system the password condition was a simple password entry prompt in the traditional style (users were asked to enter the password twice to confirm its correctness). By comparison, since we had elected to allow users to design their own cueblots we implemented a cueblot designer as part of the registration process. We found that this resulted in users spending considerable time designing their cueblot inflating the registration time (seconds) for the cueblot condition (M: 256.03, SE: 71.364) so that it was much higher than password registration time (M: 44.88, SE: 8.68), $t(52) = -2.729$, $p<0.05$. This can be viewed as a positive or negative effect depending on the reader's point of view. It clearly makes the registration more interactive, which is a good thing and is likely to lead to more memorable passwords, but, it does, unfortunately, significantly increase registration time.

We continue our discussion of time by considering the mean length of time (seconds) required to login for successful sessions. This measurement was taken from the entry of user name until the point in time when the login session was completed and may also include more than one login attempt, if users were unsuccessful at first. We found that there was a significant difference between cueblots (M: 13.08, SE:0.532) and passwords (M: 11.15, SE:0.469), $t(774) = -2.724$, $p<0.05$.This value includes any additional time it would have taken for the user's browser to download and display the image representing the inkblot. During the course of the experiment there were a total of 388 login sessions for cueblots and 412 login sessions for passwords. Of these there were a significantly lower number of sessions with a login failure for the cueblot condition (23) than for the password condition (44), $p<0.05$.This puts the mean number of failed sessions for passwords at 11% and cueblots at 6%.

However, when we look at the failed sessions in more detail we discover that within a login attempt session the average number of attempts at getting the password correct per session is somewhat different. We found that there was an average of 1.18 attempts (SE: 0.118) per session for a password while cueblots required an average of 1.96 attempts (SE: 0.493). Our results indicate that there was borderline significance ($t(65) = -1.985$, $p=0.051$)and may warrant further investigation. Thus cueblot users are more likely to get it right first time but may make more attempts to login if they fail the first time.

We also considered the number of sessions which were regarded as "total failures" ie. sessions within which there was a failed login attempt (or a sequence of failed login attempts) but no eventual success indicating that the user gave up. We found that there was no significant difference in this respect (3 failed cueblot sessions, 2 failed password sessions, $p>0.05$).

The questionnaire revealed that while most users felt that they understood how to use the cueblots, at least half found it hard to describe their cueblot, and to retain their description. They were also skeptical about the security of the mechanism.

When we embarked on this research we did not anticipate that the cueblots would not only impact negatively on usability but also not really enhance the strength of the chosen passwords. There is an implicit understanding that any authentication mechanism teeters between security and usability and a weakening of

the one will lead to a strengthening of the other. Unfortunately this was not the case when cueblots were used. In retrospect we should perhaps have anticipated this, since users will always minimise effort and maximise convenience. It is possible that this effect will disappear if we use the cueblot as a replacement for the challenge question rather than a password cue. One could go through a process of cueblot naming at enrolment and encourage the user to provide a lengthy and secure description. Users might well be willing to put the effort in since they will know that they do not have to enter this description again unless the password is forgotten.

6. CONCLUSION

In previous research into the best image for cueing purposes, we identified an inkblot-like image called a cueblot as the image that users remembered the best and which gave us the best descriptors. We then launched a website which allocated users into one of two conditions – password or password cued by a cueblot. This paper reported the findings after the website had been used for 9 weeks. In essence, what we found was that the presence of the cueblot did not really have a positive effect on the users in terms of password length or strength. One conclusion is that users simply do not need cues in a password situation. Humans famously minimize effort and therefore see security as a hurdle to be overcome. They do not want assistance in getting over a big hurdle – they want the hurdle to be as small as possible.

We still believe that the cueblots offer some potential in terms of offering users a cue when they need it. To avoid this being abused by potential intruders we advise that the user be informed either by email or by means of an SMS to their mobile phone when a cue is requested so that potential intrusion attempts can be detected immediately.

REFERENCES

[1] K Renaud, A McBryan, P Siebert. An Investigation into the Use of Images as Password Cues. Submitted for Review. 2008

[2] D Besnard and B Arief. Computer security impaired by legitimate users. *Computers and Security*, 23(3):253–264, may 2004.

[3] Sacha Brostoff and M. Angela Sasse. Are passfaces more usable than passwords?: A field trial investigation. In *Proceedings of HCI2000*, pages 405–424, 2000.

[4] H Ebbinghaus. Uberdas Gedachtnis, *Untersuchungen zue experimentellen Psychologie*. Wissenscaftliche Buchgesellschaft, 1992. reprint from 1885.

[6] Shirley Gaw and Edward W. Felten. Password management strategies for online accounts. In *SOUPS'06:Proceedings of the second symposium on Usable privacy and security*, pages 44–55, New York, NY, USA, 2006. ACM Press.

[7] M Hertzum. Minimal-feedback hints for remembering passwords. *Interactions*, pages 38–40, may-june 2006.

[8] M Moscovitch and F I M Craik. Depth of processing, retrieval cues and uniqueness of encoding as factors in recall. *Journal of Learning and Verbal Behavior*, 15(4):447–458, 1976.

[9] L Nyberg, R Habib, A R McIntosh, and E Tulving. Reactivation of encoding-related brain activity during memory retrieval. *PNAS*, 97(20):11120–11124, 2000.

[10] D L Schacter. *The Seven Sins of Memory. How the Mind Forgets and Remembers*. Houghton Mifflin Company, 2001.

[11] Richard E. Smith. *Authentication: From Passwords to Public Keys*. Addison Wesley, 2002.

[12] A Stubblefield and D Simon. Inkblot authentication. Technical Report MSR-TR-2004-85, *Microsoft Research*, August 2004. ftp://ftp.research.microsoft.com/pub/tr/TR-2004-85.pdf.

[13] A Surnelland L Zekri. Replacing passwords: in search of the secret remedy. *Network Security*, pages 4–8, January 2006.

[14] R J. Witty and K Brittain. *Automated password reset can cut IT service desk costs*, 2004. Gartner Report.

[15] James M. Wood. *What's Wrong with the Rorschach? Science Confronts the Controversial Inkblot* Test. Jossey-Bass Wiley, 2003