

Love and Authentication

Markus Jakobsson
 Palo Alto Research Center
 Palo Alto, CA 94304
 markus.jakobsson@gmail.com

Erik Stolterman
 Indiana University
 Bloomington, IN 47408
 estolter@indiana.edu

Susanne Wetzel, Liu Yang
 Stevens Institute of Tech.
 Hoboken, NJ 07030
 {swetzel,lyang}@cs.stevens.edu

ABSTRACT

Passwords are ubiquitous, and users and service providers alike rely on them for their security. However, good passwords may sometimes be hard to remember. For years, security practitioners have battled with the dilemma of how to authenticate people who have forgotten their passwords. Existing approaches suffer from high false positive and false negative rates, where the former is often due to low entropy or public availability of information, whereas the latter often is due to unclear or changing answers, or ambiguous or fault prone entry of the same. Good *security questions* should be based on long-lived personal preferences and knowledge, and avoid publicly available information. We show that many of the questions used by online matchmaking services are suitable as security questions. We first describe a new user interface approach suitable to such security questions that is offering a reduced risks of incorrect entry. We then detail the findings of experiments aimed at quantifying the security of our proposed method.

ACM Classification Keywords

H.5 Information Interfaces and Presentation; K.6.5 Security and Protection - Authentication

Author Keywords

Security question, entry error, password, reset, security

INTRODUCTION

One of the more frequent interactions that people have with computers and services starts with an authentication process. While this can be handled in many ways, the most common one is through the use of passwords. It is a widely believed fact that users are not good at keeping and remembering passwords. It is also clear that this fact in many cases leads users to use simple or *bad* passwords, or keep the same password for all situations and services. The harder people try to avoid the vulnerabilities associated with poorly chosen passwords, the higher is the risk they fail to remember their password. In this paper we present a study of a new approach to handle situations in which users forget their passwords. It is an approach that is based on insights from the fields of human computer interaction and security.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI 2008, April 5 - 10, 2008, Florence, Italy.

Copyright 2008 ACM 978-1-60558-011-1/08/04...\$5.00

The design of password authentication procedures has not developed much over the last few years. This, especially, is true for the issue of *password reset*. Password reset is a security problem of significant practical dimension. The average cost of performing a password reset involving a help desk call is estimated at \$22 [7] which is economically infeasible for many service providers. Two principal alternatives are common involving either *access* to another resource or *knowledge* of some other personal information. The former approach—in which users can request to have information sent to previously registered email addresses to enable access—is practical as long as users have access to the accounts to which the recovery information is sent but suffers from security problems associated with the delivery of this information and unauthorized access. Yet, the latter approach is vulnerable to guessing attacks both due to the inherently low *entropy* of many of the security questions used and due to the common availability of public information allowing an attacker to make educated guesses. A *combination* of the two approaches inherits the benefits of both of them, but remains problematic in the context of users who no longer have access to the email accounts to which information will be sent. This issue is particularly troublesome in situations where access is highly infrequent, as is commonly the case for some types of investment accounts such as retirement savings accounts. Some financial service providers address this practical problem by allowing users to register new email accounts after a client has proven knowledge of some recent transactions. Thus, the security of these systems is equivalent to solutions relying on knowledge alone.

Focusing on the approach involving knowledge, it is well-known that many security questions deployed to date introduce security vulnerabilities. The question *What is your favorite sports team?*, for example, has low entropy and the answer strongly depends on geographic locality, while the question *What is the name of your first pet?* is not secure if the answer is among the most common pet names (see, e.g., [2]). Also, the question *What is your mother's maiden name?* can more or less only be used in financial settings due to its historical prevalence there and suffers from vulnerabilities associated with mining of public records [4]. Some such databases also contain birth records, which are useful in determining the answers to *What is your place of birth?* It is well-known that the power of Internet search engines has led to increased possibilities for retrieving information, in some cases even information that individuals are not aware of that it is public. Recent findings [8] demonstrate the power of In-

ternet search engines to retrieve redacted or absent information. As attackers become increasingly motivated and capable, we fear that any system (e.g., [12]) based on information that is publicly accessible poses a vulnerability.

At the same time as many answers are easy for attackers to derive or guess, a second problem is that many questions may also have *many* correct answers, out of which only one will be accepted by the system. This makes the use of the system difficult and frustrating for legitimate users. For example, if a user does not recall whether he entered *Brooklyn, New York City, or NYC* as the answer to *What is your place of birth?*, then he is likely to make a mistake when having to provide this answer again. While the use of birth dates and portions of social security numbers avoids this problem, the fact that financial service providers rely on these may (justifiably so) fuel privacy concerns when other service providers (e.g., [13, 6, 11]) use these questions. This strategy may also pose liability issues in terms of the safekeeping of data.

In the following we propose and evaluate a class of new questions, dubbed *personal security questions*. These questions are chosen to relate to *personal preferences* rather than demonstrated actions and thereby avoid attacks based on data mining of public data to a large extent. Our security questions also deserve being called *personal* as they are derived from questions used to classify people placing or accessing personal ads managed by online dating services; we use these questions, strongly believing that many of them are designed to reflect long-term characteristics rather than short-term preferences¹. Our questions overcome the vulnerabilities associated with low entropy by their mere quantity. While it is straightforward to achieve a high entropy using collections of any type of security questions, we reach this goal without a notable impact on usability. This is done using a large number of multiple-response questions, from which only a relatively small portion needs to be answered. Our technique is founded in a behavioral study whose insights allow our solution to be highly resistant to a reasonable number of errors likely to be made during legitimate authentication attempts, while severely punishing the type of errors that only a stranger will make. The underlying insight is that when responding on a 3-point Likert scale—i.e., *Really like; Don't care / Don't know; Really dislike*—some responses of legitimate respondents will be off from their previously stated responses *by one point*, but *almost none by two points*. This insight is founded in the field of psychology, where it is commonly believed that *preferences* are highly stable over extended periods of time, both in comparison to short-term and long-term memory [3, 5, 9, 1].

¹One may have concerns that public information on dating web sites can be used to correctly answer the personal security questions. We note that this is highly unlikely due to the fact that while the profile is publicly available, the contact information typically is not. Thus, an attacker generally has no way to tie the answers to a specific user name. Also, the proposed personal security questions are a combination of questions taken from several dating web sites. There is no site that uses all of our security questions and it is highly unlikely that a user's authentication questions would match the online dating profile of the same user.

To assess the likely false positive and false negative rates of our proposal, we performed a series of experiments. The first one measures the entropy associated with each question; a second experiment determines the stability of subjects' preferences on the questions we chose²; a third experiment assesses the success rates of an adversary with knowledge of the probability distributions for the questions. Two types of adversaries are considered herein: strangers and acquaintances. The *stranger-adversary* can be assumed to know all frequency statistics of the answers to the questions and can make guesses that maximize his chances on average. Possible parameter choices allow a system configuration with a 0.0% false negative rate and false positive rates of 3.8% for a stranger and 10.5% for a friend.

DESIGN PRINCIPLES

It is well-known that people have problems with being creative when it comes to inventing passwords. The same seems to be true when users are given the chance to come up with their own questions. Habit and stress might lead them to re-use common questions that they have seen before which means that they also re-use the answers.

Due to its frequency and importance, the password procedure is a significant part of people's everyday interaction with computers. It is also a situation that involves many of the traditional HCI design questions. That is, questions concerning ease of use, time of use, simplicity, and of course, efficiency. The design of the questions used, the number of questions, and the form of the questions are of importance, but from an interactive point of view maybe the most important issue is how much overall demand the process puts on the user. The design should make the interaction easy and self-explanatory. Finally, the design of the questions should ideally be done in a way that both ensures high security and minimizes the reliance on externalized knowledge (such as written material, numbers, facts).

The notion of personal security questions addresses all those concerns. Our experimental findings indicate that subjects answer ten questions (all of which are prefilled) in less than 20 seconds on average. Depending on the security requirements, we estimate that between 10 and 90 questions would be used to authenticate a user.

PREFERENCE-BASED SECURITY QUESTIONS

The Authentication Approach

Our preference-based security questions approach works in two phases, *setup* and *authentication*. During the setup phase, i.e., when registering his account, a user is asked to answer a large number of questions that are related to TV programs, food, music, sports, etc. Examples include *Do you like game shows?* or *Do you like country music?*. The user is asked to respond to these question by selecting either *Really like* or *Really dislike*, or to leave the preselected answer *Don't care / Don't know* unchanged. The answers are

²The first and second experiments could have been combined into one, but for reasons related to maximizing statistical significance in the context of the available subject pools, we performed two separate experiments instead.

submitted to an authentication server. It is assumed that the submission and storage of the answers is done securely.

During authentication, i.e., when a user forgot his password, the server presents the user with a subset of the questions he was originally asked during setup, where the size of the subset determines the level of security that can be obtained. The size can be selected depending on the situation and the risk assessment made by the service provider. The answers provided during authentication are compared to the respective data stored on the authentication server. In order for the authentication to succeed, a user is allowed to make *some* errors—but not too many. In particular, the concept distinguishes between *small* and *big* errors where big errors account for dramatic changes in answers and small errors correspond to minor deviations in the answers provided. Specifically, a small error accounts for a user having a strong opinion (e.g., *Really like*) during one phase, but having no strong opinion (*Don't care / Don't know*) during the other phase, or the other way around. A big error occurs when a user has opposing strong opinions during the two phases, e.g., he answered *Really like* for a specific question during setup, but during authentication he answered *Really dislike*. While it is possible for a legitimate user to make some small errors, it is highly unlikely that a user will make a lot of big errors, considering the fact that the questions reflect a person's long-term preferences, which are relatively stable over an extended period of time. In turn, it is expected that an illegitimate user is very likely to make many big errors because he can only guess for which questions the legitimate user may have strong opinions and what the correct answers would be. These claims are experimentally supported, and the detailed findings are described in a later section.

Whether or not the authentication succeeds is based on whether or not a corresponding *score* is above or below a certain threshold. In particular, having the same strong opinion for a question in both phases will increase a user's overall score. Making a big error for a question will result in a substantial decrease of the overall score. Making a small error will neither increase nor decrease the score. Similarly, having recorded *Don't care / Don't know* as the answer during the setup phase and later answering this question correctly will neither increase nor decrease the score. (This selection is given a zero score to avoid that an adversary always selects this answer, in an effort to avoid making big errors.)

How Can One Find Good Questions?

The metric of entropy is used to determine whether a candidate question is *good* or not. We use the approach described in [10] to estimate the entropies of all candidate questions. For example, *Do you like country music?* was considered a good question because it has an entropy of 1.57, which is a high value considering the overall range [0.61, 1.57] of entropies determined in our experiments. In contrast, *Do you like to watch TV?* turned out to have a very low entropy, and thus was not selected. Only questions with high entropy are used for user authentication purposes as these are questions for which it is more difficult for an attacker to guess the correct answers.

Experiments

In the first experiment, 423 college students were asked to provide their answers anonymously to 193 questions selected from dating web sites. The students had to choose either *Really like* or *Really dislike*, or leave the prefilled answer *Don't care / Don't know* unchanged. The frequency distribution of the answers for each question was computed from the submitted answers, and it was used to estimate the probability that a common user will choose a specific option as his answer to a question. The entropy of the questions was computed based on the estimated probabilities.

The second experiment simulated the process of authentication in which 96 of the 193 questions with high entropy were used. The entropies of the 96 questions used range from 1.35 to 1.57. The experiment includes two phases: *setup* and *authentication*. During the setup phase, each subject was asked to provide his answers to the 96 questions. A user was asked to perform the authentication phase by answering the same set of the 96 questions 7-14 days after he completed the setup phase. Two instances of this experiment have been conducted: In the first instance, 46 subjects were asked to complete the setup and authentication phases receiving a \$5 reimbursement for their effort. In the second instance, which involved 26 subjects, a user starting to perform the authentication phase was informed about the possibility to win an additional \$5 in case his answers matched very well with the answers provided during the setup phase. The purpose of the second instance was to observe whether a user can do better when presented with an incentive.

In the third experiment we tested how likely it is that a user can be impersonated by strangers or acquaintances, i.e., the purpose was to evaluate the false positive rates of the authentication approach for different types of adversaries. We modeled a stranger-adversary by a machine adversary—named *Abot* (adaptive robot). The Abot guesses the answers of questions based on the known frequency distribution (as established in the first experiment). For a specific question, the Abot selects the option having the highest frequency as its answer to impersonate the targeted user. The Abot is allowed to make 1, 5, or 100 tries for the impersonation, during which it guesses the 1, 5, respectively 100 most likely collections of answers. To assess the likelihood of acquaintances succeeding in impersonating a user, we had subjects acting as adversaries to impersonate friends by trying to provide correct answers. For each authentication attempt we assign a score based on the number of correct answers with strong opinions and the number of small versus large errors. The different aspects are given different weights, where simulations are used to establish optimal parameter choices. Details of this process are beyond the scope of this publication and we refer the reader to <http://www.i-forgot-my-password.com> for more information on this matter.

What are the Error Rates?

The goal of our experiments is to find the optimal values for the parameters to minimize the likelihood to reject a legitimate user (false negative) and that of admitting an ille-

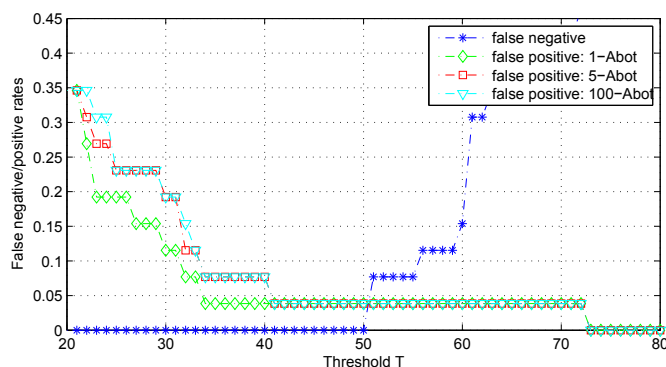


Figure 1. This figure relates to the second experiment where a \$5 incentive was offered to the subject. The x-axis shows the threshold T required to succeed with an authentication attempt. This figure shows that the more times an abot tries, the higher is its success rate (corresponding to the false positive rate). As the threshold T ranges from 41% and 50%, the error rates reach a suitable tradeoff with a false positive rate of 3.8% and false negative rate of 0.0%.

gitimate user (false positive). In the following, the parameter T denotes the threshold of the score to accept a login. Figure 1 shows that one of the optimal numerical solutions we found is for $T = 50\%$, which results in a false negative rate $fn = 0.0\%$ and a false positive rate $fp = 3.8\%$ for the Abot adversary. Aside from the results documented in Figure 1, our experiments show that providing the subjects with an extra \$5 incentive results in a decrease of the error rates by roughly 5%. Furthermore, the false positive rate for acquaintance-adversaries is 10.5% in case of $T = 50\%$.

Use of fewer questions

While the use of all 96 questions results in low error rates, our experiments show that using that many questions is unnecessary. Any subset of questions used during setup can be used during authentication. A simulation technique is used to investigate the relationship between the size of the question set and the resulting error rates. In our simulation, two factors are investigated—the *number* and *combination* of questions. Both factors have a significant impact on the resulting error rates and we find that different combinations of questions lead to different error rates. Figure 2 shows the lowest error rates we found for a given a simulation with 50 random samples of fixed-size subsets of the 96 questions. When subsets with at least 16 questions are used, the resulting error rates are tolerable, and for subsets of size 24 or greater they are very low. An extension of our approach (see <http://www.i-forgot-my-password.com>) achieves a false positive rate below 1% and a false negative rate of 0%.

CONCLUSIONS

We proposed a preference-based authentication approach in the case a user forgot his password. One main consideration in the design of our approach was to create an interaction session that puts as little as possible demand (in terms of time, memory, effort) on the user. This criterion obviously conflicts with the goal to achieve a suitable level of security. Yet, our experiments show that our approach allows to strike

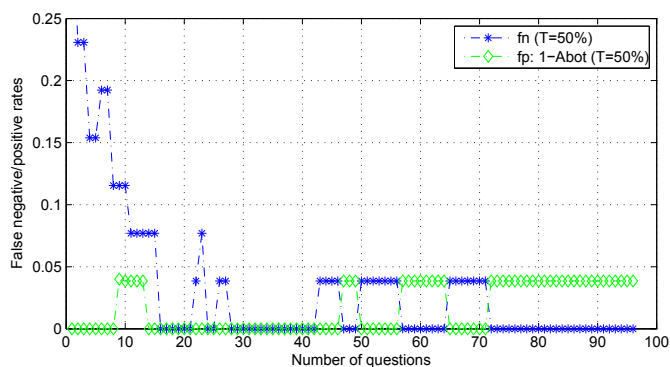


Figure 2. This figure relates to the third experiment in the case of stranger-adversaries. It shows that using very few questions results in high false negative rates, while the false positive rates keep relatively low and stable for different numbers of questions. As the number of questions increases, the resulting false negative rates decrease. Using more than 23 questions results in low and relatively stable false negative and false positive rates with values of 0.0%, respectively 3.8%.

a good balance. That is, our approach provides for low error rates while at the same time it does not ask the user for elaborate interactions that either take too much time or effort. The approach is easy to understand and fairly quick to go through, and the users in our experiments did not find the interaction intimidating or troublesome.

ACKNOWLEDGEMENTS

The authors would like to thank Prof. Susan Schept for her helpful discussions on the stability of preferences as well as friends and colleagues for helpful discussions and advice.

REFERENCES

1. K. W. Chapman, K. Grace-Martin, and H. T. Lawless. Expectations and Stability of Preference Choice. *Journal of Sensory Studies*, Vol 21(4):441–455, August 2006.
2. <http://www.bowwow.com.au/top20/index.asp>.
3. D. W. Crawford, G. Godbey, and A. C. Crouter. The Stability of Leisure Preferences. *Journal of Leisure Research*, 18:96–115, 1986.
4. V. Griffith and M. Jakobsson. Messin' with Texas, Deriving Mother's Maiden Names Using Public Records. *RSA CryptoBytes*, 8(1):18–28, 2007.
5. G. F. Kuder. The Stability of Preference Items. *Journal of Social Psychology*, pages 41–50, 10 1939.
6. Oracle Identity Management. http://www.oracle.com/technology/products/oid/oidhtml/sec_idm_training/%html_masters/c_page07.htm.
7. <http://www.voiceport.net/PasswordReset.aspx>.
8. J. Staddon, P. Golle, and B. Zimny. Web-based Inference Detection. In *USENIX Security*, pages 71–86, Boston, USA, August 2007.
9. A. E. I. Stamps. Of Time and Preference: Temporal Stability of Environmental Preferences. *Perceptual and Motor Skills*, Vol 85(3, Pt 1):883–896, December 1997.
10. D. Stinson. *Cryptography: Theory and Practice*. CRC Press, 3rd edition, November 2005.
11. Pennkey Challenge-response Password Reset Authenticating (Identifying) Yourself. https://galaxy.isc-seo.upenn.edu:7778/pls/com8i/Challenge_Controller_pg.Start_Challenge.
12. RSA Identity Verification from Verid. <http://www.rsa.com/node.aspx?id=3347>.
13. <http://www.zazzle.com/>.