Human Selection of Mnemonic Phrase-based Passwords

Cynthia Kuo
Carnegie Mellon University
cykuo@cmu.edu

Sasha Romanosky
Carnegie Mellon University
sromanos@cmu.edu

Lorrie Faith Cranor
Carnegie Mellon University
lorrie@cmu.edu

ABSTRACT

Textual passwords are often the only mechanism used to authenticate users of a networked system. Unfortunately, many passwords are easily guessed or cracked. In an attempt to strengthen passwords, some systems instruct users to create mnemonic phrase-based passwords. A mnemonic password is one where a user chooses a memorable phrase and uses a character (often the first letter) to represent each word in the phrase.

In this paper, we hypothesize that users will select mnemonic phrases that are commonly available on the Internet, and that it is possible to build a dictionary to crack mnemonic phrase-based passwords. We conduct a survey to gather user-generated passwords. We show the majority of survey respondents based their mnemonic passwords on phrases that can be found on the Internet, and we generate a mnemonic password dictionary as a proof of concept. Our 400,000-entry dictionary cracked 4% of mnemonic passwords; in comparison, a standard dictionary with 1.2 million entries cracked 11% of control passwords. The user-generated mnemonic passwords were also slightly more resistant to brute force attacks than control passwords. These results suggest that mnemonic passwords may be appropriate for some uses today. However, mnemonic passwords could become more vulnerable in the future and should not be treated as a panacea.

Categories and Subject Descriptors

K.6.5 [**Computing Milieux**]: Security and Protection – *Authentication*; H.1.2 [**Models and Principles**]: User/Machine Systems – *Human Factors*

General Terms

Security, Human factors

Keywords

Mnemonic phrases, password selection, password cracking, user studies.

1. INTRODUCTION

Today's computer users manage a large number of online accounts that require passwords. Each system may have different rules for what passwords are acceptable and what passwords are not. Some passwords must be under eight characters; some must be over eight characters; some must contain multiple classes of characters; some cannot accept certain characters. It is no wonder that users have difficulty creating and remembering strong passwords.

Some systems try to make password selection easier by instructing users to create mnemonic phrase-based passwords [2] [6] [20] [31] [32] [33] [34]. (In this paper, we also refer to mnemonic phrase-based passwords as mnemonic passwords.) Users are instructed to create mnemonic passwords with directions similar to the following:

> Choose a password that is hard for other people to guess but easy for you to remember by doing the following:
>
> 1. Think of a memorable sentence or phrase containing at least seven or eight words.
> 2. Select a letter, number, or special character to represent each word in your password. A common method is to use the first letter of every word.
> 3. Ideally, the password should contain a mixture of lower case and upper case letters, numbers, punctuation, and special characters (such as ^ or %).
> 4. Remember the phrase.

Table 1 shows examples of mnemonic phrases, their derived passwords, and the inspirations for these phrases.

Table 1: Examples of Memorable Phrases and Passwords

Phrase	Password	Inspiration
Four score and seven years ago, our Fathers	4s&7yaoF	Quotation – Gettysburg Address
I love to ski at Seven Springs!	Ilts@7S!	Personal – Hobby
Alas, poor Yorick! I knew him, Horatio	A,pY!Ikh,H	Literature - "Hamlet" by Shakespeare

It is often assumed that mnemonic passwords will be stronger than "regular" passwords (i.e., passwords created without specific instructions) for three reasons. First, mnemonic passwords do not appear in any password cracking dictionary. Second, the phrases will help users incorporate different character classes, such as upper case letters or punctuation, into their passwords. Last, the space of possible phrases is virtually infinite. However, there is little empirical data on the quality of mnemonic passwords that users select in practice. We simply do not know whether mnemonic passwords are as strong as commonly believed.

In this paper, we discuss our assessment of mnemonic password strength, relative to regular passwords. First, we expect that many users will choose phrases that have been posted on the Internet, such as music lyrics, advertising slogans, or famous quotations. As a result, it is possible to build a dictionary for mnemonic passwords. We demonstrate this with a proof of concept. Second, we conduct an online survey to gather human-generated mnemonic and regular passwords. These passwords are then run against the corresponding dictionary and scored on their strength. To score the passwords, we use a system based on: length, number of character classes, and presence in a cracking dictionary. Our analysis suggests that instructing users to create mnemonic passwords may not necessarily strengthen passwords against automated crackers. In fact, mnemonic passwords have the potential to be even *more* vulnerable than regular passwords.

2. RELATED WORK

In this paper, we focus solely on the creation of textual passwords. Researchers have developed many alternative schemes for authentication, such as biometrics, one-time passwords, and graphical passwords. While these alternatives are promising, systems will continue using textual passwords for many years.

Yan et al. evaluate the security and memorability of "regular" passwords, mnemonic passwords, and random passwords [36]. They conclude that mnemonic passwords are much stronger than regular passwords and as strong as random passwords. However, their analysis relies on a standard (non-mnemonic) crack dictionary to measure the strength of mnemonic passwords.

More generally, studies on password selection, memorability and usability conclude that people choose poor passwords [1] [4] [28]. Users tend to choose short passwords and derive them from personal information that is easily guessable. Users must also manage many passwords and often reuse a password across different accounts.

2.1 Alternative Textual Password Schemes

In cognitive password authentication, the system randomly selects a set of personal questions each time the user logs in [37]. Cognitive passwords have high recall rates, but they may be impractical for widespread use. The factual- and opinion-based questions may be easy for family or friends to guess. Also, every organization would need a unique set of questions to prevent reuse [3].

Pass-sentences and pass-phrases are textual passwords composed of long, grammatically correct phrases [29]. The personalized phrases are memorable, and their length resists software cracking. However, the increased length makes them impractical for repeated use [3].

Randomly generated, human pronounceable passwords are produced by concatenating pronounceable syllables into new "words." Most pronounceable password generators [7] [35] are based on Morrie Gasser's work [9] [11]. While the algorithm generates passwords that resist standard dictionary attacks, there are weaknesses in the Gasser algorithm [10].

Other systems generate mnemonic phrases for passwords [16]. Given textual passwords, these systems return grammatically correct phrases that users can use as memory aids for their passwords. The memorability of these system-generated phrases is untested, but user-generated passwords are generally more memorable than system-generated passwords.

3. ATTACKS AGAINST TEXTUAL PASSWORDS

Attackers generally compromise passwords in one of four ways:

1. By gathering enough information about users to guess their password;
2. By social engineering, e.g., tricking users into revealing their usernames and/or passwords;
3. By capturing users' passwords, e.g., via shoulder surfing or spyware; and
4. By cracking passwords using a software program, such as John the Ripper.

Attacks 2 and 3 are both significant threats, but they are not affected by human password selection. Attack 1 is affected by password selection to the extent that users create passwords with available personal information. This paper focuses on Attack 4, securing passwords against automated password cracking.

Attackers currently have two methods for automated password cracking. First, they employ a dictionary,[1] which is a list of common words. Standard password dictionaries contain known passwords, such as people's names, pets' names, locations, and names of sports teams. More comprehensive dictionaries are also available online [23]. Dictionaries are available in numerous languages (not just English), and there are domain-specific lists as well.

Cracking software tests each word in the dictionary against the targeted password. The attacker can also permute words so that common modifications will also be tested. For example, John the Ripper will append numbers to words, change lower case letters to upper case letters, and substitute numbers for similar-looking letters (e.g., '3' for 'E'). The permutations are not comprehensive, but attackers can modify the rules as needed.

If dictionaries are unsuccessful, attackers can use brute force attack. A brute force attack attempts to match every possible combination of characters against a password. Since this is time-consuming, cracking software optimizes brute force attacks using character frequency tables. Frequency tables exploit an innate property of language: certain characters occur more frequently than others. Clearly, attackers want to use frequency tables that best match the targeted passwords – but what would this be?

A perfectly random set of passwords would produce a uniform distribution of characters. However, humans rarely generate random passwords. Character frequencies in users' languages will affect their passwords. However, the character frequencies in typical usage may not represent the characters used in passwords; for example, many of the commonly used characters in English occur in articles and prepositions. These words appear in passwords less frequently than in written English. Instead, the frequencies of characters in a password cracking dictionary would better reflect the composition of actual passwords. The frequencies from John the Ripper's 1.2 million-word English dictionary are shown in Figure 1.

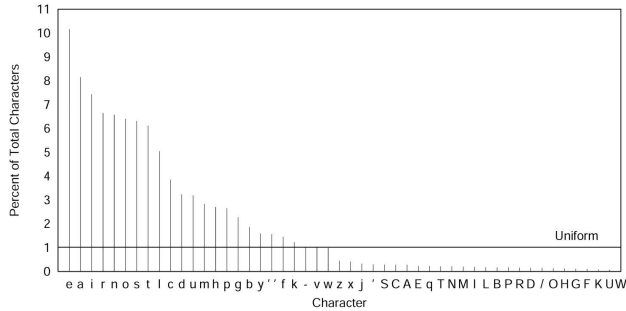[1] John the Ripper refers to dictionaries as "wordlists."

Figure 1: Frequency of 50 Most Common Characters in John the Ripper's Password Dictionary

4. SURVEY METHODOLOGY

Although advice on creating mnemonic passwords is widespread [2] [6] [20] [31] [32] [33] [34], there is limited empirical evidence of its effectiveness. To assess the mnemonic passwords that users create, we conducted a survey to gather a sample of mnemonic and control (i.e., minimal creation instructions provided) passwords. The survey ran for 15 days in February, 2006. Respondents were recruited by messages posted on online bulletin boards, including:

- The "volunteers" section on Craigslist, a free bulletin board service that offers localized listings for various cities.
- The "volunteers" section on backpage.com, which is similar to Craigslist.
- A student bulletin board hosted at our university.

The recruiting message specifically asked for individuals who are over 18, have password-protected accounts at 5 or more websites, and have made at least one purchase online. The message directed respondents to a server at our university. The server randomly redirected respondents to one of two surveys hosted at www.surveymonkey.com.[2] In both surveys, we asked participants to create a password. The following recommendations were provided but not enforced:

- Passwords should be at least eight (8) characters in length.
- Passwords should contain a mixture of lower case and upper case letters, numbers, punctuation, and special characters (such as ^ or %).

Because we wanted to judge the quality of the passwords that participants created, we needed access to participants' cleartext passwords. Thus, we instructed participants *NOT* to provide a password (or a variant of a password) that they currently use or have previously used for another account.

One survey (the control) did not provide instructions on password creation beyond the two recommendations provided above. The second survey (the experimental condition) instructed participants to create a mnemonic phrase-based password. The instructions for creating a mnemonic password are included in Section 1. In both surveys, we also asked participants about their password selection

---

[2] Survey Monkey is a website that automates online survey administration.

and management behavior. The survey questions can be found in Appendices A and B.

In exchange for their time, participants were entered into a raffle for an iPod Nano. They were instructed to return to our university site after the survey closed – and login using the password they created – to check if they had won the iPod. Roughly sixty percent returned to our site.

By advertising on online bulletin boards, we sought a study population that is *more* web-savvy than the average Internet user. These users should create better passwords than the average user. Therefore, the results we present reflect a worst-case scenario for attackers; conversely, the results reflect a best-case scenario for system administrators and researchers.

5. SURVEY ANALYSIS

We analyzed the passwords generated by our survey participants using several methods. First, we tried cracking the passwords. The control passwords were matched against John the Ripper's English dictionary, and the mnemonic passwords were matched against a computer-generated mnemonic dictionary (described in Section 5.1.1). Next, we used the results from the cracking exercise to score the strength of the passwords. Finally, we estimated an upper bound for the effectiveness of the mnemonic dictionary. This was accomplished by searching online for the phrases that respondents used to generate their mnemonic passwords.

5.1 Password Cracking

We employed three different methods to crack the passwords:

- **Basic Dictionary Attack**: UNIX password files were generated from the cleartext passwords, and John the Ripper was invoked using the "-wordlist" command line option. This option checks the password against each word in the dictionary. Control passwords were tested using a dictionary [23] of about 1.2 million words. Mnemonic passwords were tested using our 400,000 word mnemonic dictionary.

- **Dictionary Attack with Permutations**: We invoked John the Ripper with the "-rules" command line option on each dictionary. This performed "word mangling" through character replacement (e.g., replacing "a" with "@"), capitalization, the addition of prefixes and suffixes, and other permutations.

- **Brute Force Attack**: John the Ripper was invoked without command line options, forcing it to try all combinations of characters. The passwords in each condition were subjected to brute force attack for 62 hours.

5.1.1 Mnemonic Phrase-based Password Dictionary

For many individuals, it is easier to recall a phrase from an existing source than to create a new phrase. Popular phrases may be derived from songs, speeches, poetry, books, plays, advertising slogans, and so on. The text to these popular phrases is often readily available on the Internet, lending itself to inclusion in a mnemonic phrase-based password dictionary.

To build our mnemonic password dictionary, we gathered phrases by screen-scraping several aggregation sites. We gathered

advertising slogans [8] [27], children's nursery rhymes and songs [26], movie quotes [15], famous quotations [21] [25], song lyrics [30], and television theme song lyrics [5]. Each site appeared within the top five search results returned by Google (for the appropriate search string). We selected sites with HTML formats that were easy to scrape. In total, we gathered over 249,000 phrases. Only 129,000 phrases generated passwords with eight or more characters.

To generate a password, we first concatenated the first character of each word in a phrase. We then produced variations using obvious word or character substitutions. Examples of these substitutions are shown in Table 2. We also generated variations that included the punctuation in a phrase.

Table 2: Examples of Word / Character Substitution Rules

Original	Replacement
Four, for, fore	4
Dollar	$
Your, you're	ur

For example, take the phrase "Two men look out through the same bars: One sees the mud and one the stars". Concatenating the first letters yields "tmlottsbostmaots". Substitutions transform the password into "2mlottsb1stm+1ts". Adding punctuation produces "2mlottsb:1stm+1ts." These rules yielded over 400,000 unique dictionary entries.

Figure 2 illustrates the character frequencies from our computer-generated mnemonic dictionary. Compared to John the Ripper's dictionary (shown in Figure 1), the distribution of characters in the mnemonic dictionary appears to be more uniform. The ten most frequent characters comprise 67% of the characters in the John dictionary, compared with 57% in the mnemonic dictionary.
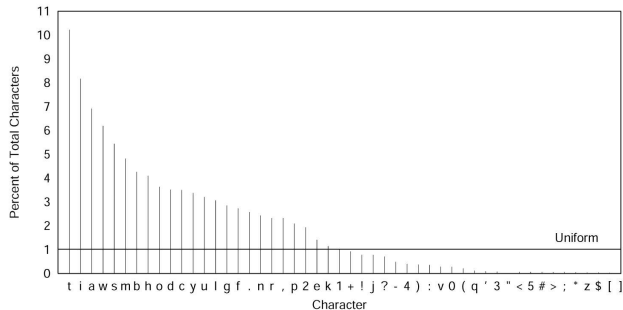

Figure 2: Frequency of 50 Most Common Characters in Computer-Generated Mnemonic Password Dictionary

5.2 Password Scoring

Password scorers can be found in numerous applications, from the Mac OS X password assistant [19] to Firefox/Thunderbird [22] to Google Accounts [12]. The underlying algorithms may differ slightly but are similar in essence: passwords earn higher scores when they are longer and when they include more character classes.

Unfortunately, all of these scoring applications provide feedback using qualitative progress bars. We wanted to quantitatively compare the relative strength of mnemonic and control

passwords. Based on existing password scorers, we developed a scoring system which incorporates three factors:

1. Whether the password is present in an existing crack dictionary;
2. The length of the password; and
3. The size of the character classes contained in the password.

Clearly, if a password can be cracked by a dictionary, it should earn a low score. Making a password longer or including more character classes increases the size of the search space necessary for a brute force attack. Thus, stronger passwords should earn higher scores.

Each password was scored as follows:

$$Score = \begin{cases} Log_{10}(\ (Num\ Characters)^{Length}\ ) & \text{Not in dictionary} \\ 0 & \text{In dictionary} \end{cases}$$

such that the *Length* is the number of characters is the password, and *Num Characters* is the total number of characters in the password's search space. *Num Characters* is determined by the character classes in the password. There are 26 upper [A-Z] and lower [a-z] case letters, 10 numbers [0-9], and 33 special characters (including punctuation, the dollar sign, asterisk, tilda, space, and so on). Thus, if a password contains lower case letters and numbers, *Num Characters* = 26 + 10 = 36. This better captures the size of the possible search space than using the number of character classes. (For example, there are fewer numbers than lower case characters. Systems that only track the number of character classes weight numbers and lower case characters equally.) The Log function reduces the score to an order-of-magnitude estimation of effort. A password compromised by a dictionary attack automatically earns a score of zero. A six-character password that is not in the dictionary and contains lower case letters and numbers scores $Log_{10}(\ 36^6\ )$ * 1 = $Log_{10}(2.2 \times 10^9)$ = 9.3.

For comparison purposes, we scored the dictionary passwords *as if they were not listed in the dictionaries*. The passwords in John the Ripper's English dictionary scored an average of 13.5, with a standard deviation of 5.6. The passwords in our mnemonic phrase dictionary, which contain more punctuation, scored an average of 15.3, with a standard deviation of 6.9. We did not score automatically generated permutations of the entries.

5.3 Availability of Mnemonic Phrases

Survey respondents in the mnemonic condition were asked to provide a base phrase as well as the password derived from the phrase. We used the Google API to query for each respondent's base phrase [13]. This test provides an upper bound on how many passwords could potentially be cracked using a dictionary. We postulate that if many of the mnemonic phrases are found in a search, then it would be possible to build an effective dictionary for cracking mnemonic passwords.[3]

---

[3] Results using this metric may be slightly inflated. Google strips punctuation from search terms, even if the terms are placed in quotations. Thus, the phrases that respondents provided may have matched phrases with the same words but punctuated differently. Over 60% of phrases contained punctuation.

6. RESULTS

A total of 290 individuals completed our survey, with 146 respondents in the control condition and 144 respondents in the mnemonic condition.

6.1 Respondents' Password Strength

We first tried to crack respondents' passwords using John the Ripper. Initially, we only tested the entries in the dictionary and simple permutations thereof. Not surprisingly, more control passwords were cracked than mnemonic passwords: 11% of control passwords versus 4% of mnemonic passwords were compromised. This was expected; the control dictionary is three times larger than the mnemonic dictionary, and it has been tailored to crack real passwords. However, the control passwords fared better than we had anticipated – Klein cracked 24% of passwords with a 63,000-word permuted dictionary, and Yan et al. cracked over 30% in a student population using a permuted dictionary attack [17] [36]. In comparison, a crack rate of 11% is surprisingly low.[4]

Next, we scored the strength of the passwords, according to the system outlined in Section 5.2. We found no significant difference between the scores of the control passwords and the mnemonic passwords (tested using one-way ANOVA, $t(292) = 1.7$, $p = 0.10$, where $\alpha = 0.05$). The mean and standard deviation for each group are shown in Table 3. The mnemonic passwords did not incorporate more character classes than the control passwords. In addition, the two groups created passwords of roughly the same length.

Table 3: Password Strength (Mean and Standard Deviation)

	Control	Mnemonic
Strength Score	15.7 ± 7.3	17.2 ± 8.3
Number of Character Classes	2.9 ± 1.0	2.7 ± 0.9
Length (Number of Characters)	9.9 ± 2.7	9.5 ± 4.1

Last, we ran a brute force attack against the remaining passwords. The brute force attack cracked an additional 8% of the control passwords and 4% of the mnemonic passwords. Note that we did *not* modify the frequency tables for the mnemonic passwords. Our results are summarized in Table 4.

---

[4] There are many possible explanations for this result. First, our survey population may be comprised of better password creators than the other study populations. Second, the studies may have used different dictionaries or permutations. Third, both Klein and Yan et al. tried attacks with personal information (such as usernames), which we did not have. Fourth, there may be an experimental bias where survey respondents created better passwords for the study, knowing the passwords would not be used regularly. Finally, the quality of passwords could genuinely be improving.
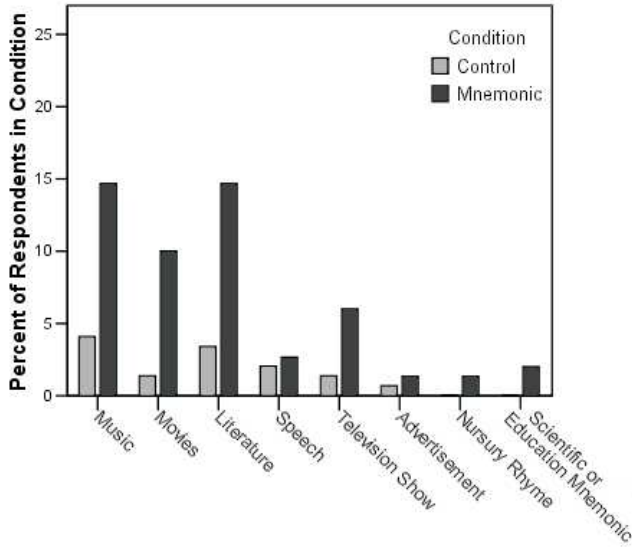
Table 4: Password Cracking Results

	Control	Mnemonic
% of Passwords Compromised by Basic Dictionary Attack	6%	3%
% of Additional Passwords Compromised by Dictionary Attack with Permutations	5%	1%
Dictionary Dictionary Size	John the Ripper 1.2 million entries	Custom (ours) 400,000 entries
% of Additional Passwords Compromised by Brute Force Attack	8%	4%
Frequency Table	Standard (John the Ripper's default)	Standard (Ideally, this frequency table should be optimized for mnemonic passwords.)

6.2 Building a Better Mnemonic Dictionary

Mnemonic phrase-based passwords will only be effective if users select phrases that do not appear in a mnemonic dictionary. (For the sake of argument, we assume that users perform predictable transformations on their phrases, such as selecting the first letter of every word.) If users select readily available phrases that are posted on the Internet, determined attackers could build a comprehensive dictionary. Even if users deviate from selecting the first character, the resulting passwords are often simple permutations. For example, one respondent created the password "Subg,1cb" by transforming the Kelly Clarkson lyric, "Since you've been gone, I can breathe." Another respondent used the *Princess Bride* movie quote, "My Name is Enigo Montoya. You killed my father. Prepare to die!" to produce the password "Mniem.Ykmf.Ptd!" These passwords contain simple permutations that already appear in John the Ripper's permutation rules.

We hypothesized that users would select phrases that are publicly available. After all, the act of recalling a phrase proves that it is memorable. It is also easier to recall a phrase than to think of a new one. Survey respondents in the mnemonic condition based their passwords on external sources more often than the control group ($\chi^2 = 53.0$, $p < .005$). In fact, the *majority* of mnemonic passwords were based on external sources, compared to only 13% of respondents in the control group. The external sources are categorized in Figure 3, and examples of passwords based on these sources are shown in Table 5.

Figure 3: Percent of Passwords Based on External Sources

Thus, a determined attacker could focus on gathering more phrases from music, movies, and literature. This has the potential to greatly increase the crack rate of the mnemonic dictionary.

Table 5: Examples of Passwords Based on Media Sources

Condition	Password	Source
Mnemonic	SWMtM$$!!	Movie: Based on the quote, "Show me the money!" from *Jerry Maguire*.
Control	atreyu09	Movie: Atreyu is a character in *The NeverEnding Story II*.
Mnemonic	iwtbotaiwtwot	Book: Based on the opening sentence from Dickens' *Tale of Two Cities*.
Control	LifeIs42	Book/Movie/Radio/TV show: Based on the answer in *The Hitchhiker's Guide to the Galaxy*.

6.3 Availability of Mnemonic Phrases

To estimate the upper bound of the effectiveness of a mnemonic dictionary, we used Google to search for the base phrase of each mnemonic password. We were able to find 65% of the phrases that respondents used by performing Google searches. (Interestingly, two different respondents used lyrics from the Oscar Meyer Weiner jingle.)

While building a comprehensive phrase dictionary is a non-trivial task, these results suggest that a dictionary could be highly effective against mnemonic passwords – unless users are trained to avoid publicly available phrases.

6.4 Frequency of Characters in Respondents' Passwords

We compared the character frequencies in respondents' passwords to what we expected based on the password dictionaries. Figure 4 shows the character frequency of respondents' passwords in the control condition. By visual inspection, this distribution appears to be much closer to uniform than the "John" distribution shown in Figure 1.[5] The most frequent characters in John the Ripper's dictionary appear in respondents' passwords *less* frequently than expected, and conversely, many of the lesser-used characters in the dictionary appear in respondents' passwords *more* frequently than expected.
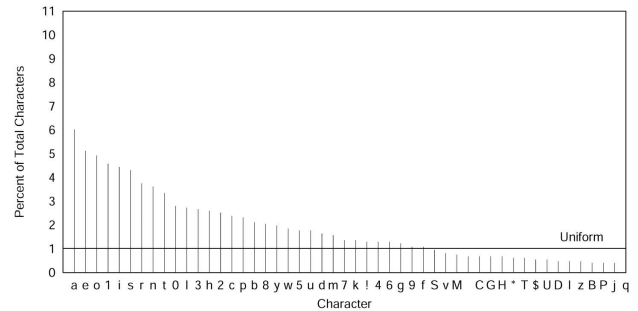

Figure 4: Frequency of 50 Most Common Characters in Control Group Passwords

Similarly, Figure 5 shows the character distribution of respondents' passwords in the mnemonic condition. Figure 5 also appears to be much closer to uniform that its counterpart, the mnemonic dictionary distribution shown in Figure 2.
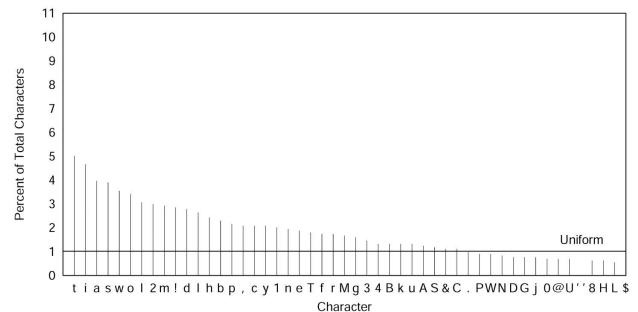

Figure 5: Frequency of 50 Most Common Characters in Mnemonic (Experimental) Group Passwords

Attackers must resort to brute force attacks for passwords that do not appear in a dictionary. Brute force attacks are most difficult when each character appears with equal probability. The highly skewed distributions in Figure 1 and Figure 2 indicate that

---

[5] We are not aware of an appropriate method for statistically comparing the two frequency distributions. For example, the standard chi-square goodness-of-fit test would be appropriate for comparing a survey or dictionary distribution to a uniform distribution. However, it was not designed to compare two observed distributions or to decide which of two observed distributions are closer to uniform.

attackers could optimize frequency tables for brute force attacks. Thus, the survey distributions shown Figure 4 and Figure 5 are extremely encouraging.

Interestingly, the control and the mnemonic passwords appear to be equally susceptible (or resistant) to brute force attacks. Visual inspection indicates the distributions in Figure 4 and Figure 5 are similar. We also calculated the percentage of passwords that could be compromised using only the top n most frequent characters. In other words, suppose an attacker limited her search space to the 30 most frequently occurring characters. If she tested every combination of those 30 characters, she could compromise some percentage of the passwords – in this case, 20% of the control passwords and 15% of the mnemonic passwords. Control passwords were tested against the order of characters in John the Ripper's dictionary, and mnemonic passwords were tested against the order of characters in our computer-generated dictionary. As shown in Figure 6, the mnemonic group performs slightly better than the control group when testing fewer than 40 characters. However, there is virtually no difference between the two groups when a larger character set is tested.
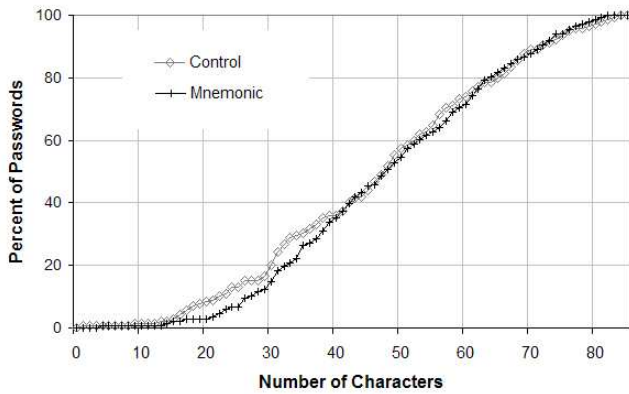

Figure 6: Percent of Passwords Comprised Solely of the n Most Frequent Characters

6.5 User Education and Experience

In our survey, we asked respondents to identify which of the following factors they considered when creating their new password:

- Does not contain dictionary words;
- Is in a foreign (non-English) language;
- Is not related to the site (i.e., contains the name of the site);
- Includes numbers;
- Includes capital letters;
- Includes special characters (for example, "&" or "!"); and
- Is long enough.

These factors often appear in educational materials about passwords or in enforcement rules during password creation.[6]

---

[6] A possible exception is the foreign language factor. However, we have noticed that many individuals use non-English words in their passwords, based on the belief that it will increase the strength of their passwords.

We also asked respondents if anyone had ever tried to explain to them "what a good password is." Three-quarters of respondents reported that they had received some type of password education. Interestingly, forty participants in the remaining one-quarter of respondents were in the mnemonic condition – and had just read about mnemonic password creation. We interpret these responses as an indicator of whether participants understood and internalized educational guidelines.

Password education does appear to affect user behavior. Respondents who said they had received password education considered an average of 3.1 factors, compared to an average of 2.1 factors with the negative responses (tested using one-way ANOVA, $t(292) = 4.5$, p < .005).

In addition, respondents who reported having more online passwords considered more factors ($t(291) = 2.4$, p < .005). This is summarized in Table 6. Respondents with more passwords presumably have encountered more attempts at password education. Internalizing password enforcement rules may also be a time-saving mechanism for experienced users. Users do not want to waste time having password candidates rejected by the system; instead, they automatically create passwords that will comply with enforcement rules.

Table 6: Number of Factors Considered during Password Creation

	# Respondents	Mean	Std Dev
0 – 10 Passwords	98	**2.5**	1.6
11 – 20 Passwords	117	**2.9**	1.5
20+ Passwords	78	**3.3**	1.7

Overall, the results indicate that password education is having some effect on users' behavior. This is extremely encouraging.

Between the control and the mnemonic conditions, there was no detectable difference in the total number of factors considered.

We also measured other metrics of online experience, and they all had no effect on the number of factors considered. Tested metrics include: the length of time that respondents had been using the Internet; the number of online purchases in the past month; and whether respondents reported that a password had been compromised.

7. DISCUSSION

7.1 Memorability

A good password is often defined as one that is memorable and hard to guess. In this paper, we focused on the latter condition: the difficulty of guessing a password depends on the quality of the password selected. Memorability is no doubt important, but it may not be a necessary condition.

Experts sometimes frown upon users who write down their passwords. However, for online accounts, writing down many unique passwords may be preferable to reusing one password across different sites [18]. Furthermore, password management systems, such as Password Safe [24], can be used to securely store passwords in encrypted form.

7.2 Future Work

During the course of our research, we found more questions than definitive answers. For example, we wondered if users can be trained to base mnemonic passwords on phrases that are not posted on the Internet. If so, we would still need to measure whether those passwords are measurably better.

It is also unclear how users will transform their phrases into passwords. Less than 60% of respondents formed their password using only the most basic transformations (i.e., using the first letters of words and standard number-for-letter substitutions). The remaining respondents used unexpected transformations. For example, one user changed "I <heart> peanut butter and jelly" to "aye<3pbnj." What other permutations will people use? Will shorthand from instant messaging and text messaging appear in passwords? Mnemonic phrase-based passwords warrant more attention from the research community; this survey only scratches the surface.

More generally, we were intrigued by the idea that users' passwords might be improving over time. It would be interesting to conduct a longitudinal study on the quality of passwords.

Also, the impact of password management systems on passwords quality is unknown. Are these systems more secure because they can be used to conveniently generate and store strong passwords?

8. CONCLUSION

When instructed to create mnemonic phrase-based passwords, the majority of users select phrases from music lyrics, movies, literature, or television shows. The text of these sources is often available on the Internet. This opens the possibility that a dictionary could be built for mnemonic passwords. If a comprehensive dictionary is built, it could be extremely effective.

Mnemonic phrase-based passwords are not as strong as people may believe, but that does not mean that we should refrain from using them. To the best of our knowledge, passwords based on mnemonic phrases do not appear in password cracking dictionaries. In addition, a relatively small percentage of users actually employ this method of password creation; the incentive to adapt password cracking software for mnemonic passwords is low. Finally, the space of possible phrases is extraordinarily large, and building a comprehensive dictionary is not a trivial task. There are also more permutations that can be made on mnemonic phrases, increasing the size of the search space. It may be possible to crack a significant percentage of mnemonic passwords in theory — but this is different from today's reality.

System designers and administrators should specifically recommend to users that they avoid generating mnemonic passwords from common phrases. (Some organizations actually suggest using phrases from common sources [32], but this should be changed.) Because mnemonic phrases lend themselves to a rich set of transformations, providing examples of these transformations may help stimulate users' creativity. While mnemonic-phrase based passwords may not be a panacea for the password selection problem, they do offer a user-friendly alternative for encouraging users to create good passwords.

10. REFERENCES

[1] Adams, A., and Sasse, M.A., 1999. Users are not the enemy: why users compromise security mechanisms and how to take remedial measures. Communications of the ACM 42 (12), 40-46.

[2] Australian Computer Emergency Response Team (AusCERT). Choosing good passwords. AusCERT Reference # GoodPasswords, February 1, 2001. http://www.auscert.org.au/render.html?it=2260 (accessed March 2006).

[3] Brostoff, S. Performance of Authentication Mechanisms. PhD Thesis, Chapter 4. http://www.cs.ucl.ac.uk/staff/s.brostoff/thesis/sachas_thesis_ch04.pdf (accessed May 2006).

[4] Brostoff, S., and Sasse M. A. Ten strikes and you're out: Increasing the number of login attempts can improve password usability. CHI 2003 Workshop on Human-Computer Interaction and Security Systems, Ft. Lauderdale, Florida.

[5] The Classic TV Database. Classic TV Theme Songs - Theme Song Lyrics. http://www.classic-tv.com/themesongs/lyrics.asp (accessed July 2004).

[6] Curry, D.A. Selecting Good Passwords. Excerpts from Improving the Security of Your UNIX System. http://www.alw.nih.gov/Security/Docs/passwd.html (accessed March 2006).

[7] Debian Package: GPW, http://www.mnis.fr/deb30/utils/gpw.html (accessed May 2006).

[8] Fact-index.com. List of Advertising Slogans. http://www.fact-index.com/l/li/list_of_advertising_slogans.html (accessed July 2004).

[9] Federal Information Processing Standards Publication 181, Standard for Automated Password Generator. National Institute of Standards and Technology, October 5, 1993. http://www.itl.nist.gov/fipspubs/fip181.htm (accessed February 2006).

[10] Ganesan, R. and Davies, C. A New Attack on Random Pronounceable Password Generators. Proceedings of the 17th {NIST}-{NCSC} National Computer Security Conference, 1994.

[11] Gasser, M. A Random Word Generator for Pronounceable Passwords. Technical Report ESD-TR-75-97, Electronic Systems Division, Hanscom Air Force Base, 1975.

[12] Google Accounts. "Edit Password." https://www.google.com/accounts/EditPasswd (accessed May 2006).

[13] The Google API, http://www.google.com/apis/ (accessed February 2006).

[14] John the Ripper, http://www.openwall.com/john/ (accessed February 2006).

[15] Johnson, B.B. The Movie Quotes Site. http://www.moviequotes.com/repository.cgi (accessed July 2004).

[16] Jeyaraman. S., and Topkara, U. Have the cake and eat it too- Infusing usability into text-password based authentication systems. CERIAS and Department of Computer Sciences, Purdue University, 2005.

[17] Klein, D. V., Foiling the Cracker; A Survey of, and Improvements to Unix Password Security", (revised paper with new data) Proceedings of the 14th DoE Computer Security Group, May 1991.

[18] Kotadia, M. Microsoft Security Guru: Jot Down Your Passwords. CNET News.com, May 23, 2005. http://news.com/Microsoft+security+guru+Jot+down+y our+passwords/2100-7355_3-5716590.html (accessed March 2006).

[19] Mac OS X Password Assistant. "Passwords: Safety in Numbers." http://www.apple.com/macosx/tips/password13.html (accessed May 2006).

[20] Microsoft Corporation. Strong Passwords – How to Create and Use Them. Security At Home, Personal Information, November 30, 2005. http://www.microsoft.com/athome/security/privacy/password .mspx (accessed March 2006).

[21] Moncur, M. The Quotations Page. http://www.quotationspage.com/quotes/ (accessed July 2004).

[22] Mozilla Corporation, http://www.mozilla.com.

[23] Password Cracking Wordlist, http://www.openwall.com/wordlists/ (accessed February 2006).

[24] Password Safe, http://passwordsafe.sourceforge.net/ (accessed February 2006).

[25] Quoteland.com. Quoteland.com…all the right words! http://www.quoteland.com/author.asp (accessed July 2004).

[26] Rhymes.org.uk. Nursery Rhymes - Lyrics and Origins! http://www.rhymes.org.uk/ (accessed July 2004).

[27] Richards, J.I. Research. University of Texas at Austin, Department of Advertising, February 10, 1997. http://advertising.utexas.edu/research/slogans/index.asp (accessed July 2004).

[28] Sasse, M. A., Brostoff, S., and Weirich, D. Transforming the weakest link: a human-computer interaction approach to usable and effective security. BT Technology Journal, Vol 19 (3), 2001, pp. 122-131.

[29] Spector, Y., and Ginzberg, J. Pass-sentence - a new approach to computer code. Computers & Security, Vol 13, 1994, pp. 145-160.

[30] The Song Lyrics. Song Lyrics. http://www.thesonglyrics.com/ (accessed July 2004).

[31] United States Coast Guard, http://www.uscg.mil/HQ/PSC/cghrms/using_peoplesoft/how _to_change_your_password.htm (accessed February 2006).

[32] University of Chicago, Networking Services and Information Technologies. "Choosing Good Passwords," 2002. http://security.uchicago.edu/docs/userpassword.shtml (accessed March 2006).

[33] University of Colorado, Department of Computer Science. "Password Policy," November 2004. http://www.cs.colorado.edu/~lizb/internal/password-policy.html (accessed February 2006).

[34] University of New Orleans, Department of Computer Science. How Do I Create A Secure Password? Reprint of article in ;login 21, no. 3 (1996). http://www.cs.uno.edu/Resources/FAQ/faq4.html (accessed February 2006).

[35] Van Vleck, T. "Java Password Generator," July 31, 1997. http://www.multicians.org/thvv/gpw.html (accessed May 2006).

[36] Yan. J., Blackwell A., Anderson, A., and Grant A. The Memorability and Security of Passwords -- Some Empirical Results. Technical Report No. 500, Computer Laboratory, University of Cambridge, 2000.

[37] Zviran. M., and Haga, W.J. Cognitive Passwords: The Key to Easy Access Control. Computers & Security, Vol 9, 1990, pp. 723-736.