MEGAMIND
Myths about Password Settings and Other Nonsense: How Information Security Tortures Users in the Name of Security

E. Eugene Schultz, Ph.D., CISSP, CISM

Abstract

Typical organizations have information security standards that require a certain password length, password expiration every 30 to 90 days, password complexity, and more. Information security staff members who routinely prescribe these settings might believe that their organization is meeting "best practice" standards. Research on password settings over the past years, however, suggests that many widely accepted and used settings do not help security appreciably. Instead, many of these settings not only inconvenience users, but in many cases make them less able to remember their passwords. The problem is not limited to passwords, either. Third-party authentication and other technology designed to improve security too often are not at all user friendly. This paper discusses how information security tortures users in the name of security and suggests solutions.

Introduction

The major goals of information security are to protect the confidentiality of information, integrity of information, systems, and applications, and availability of information, systems, and applications. In the pursuit of these goals information security practices select and implement three major types of controls, technical, physical and administrative. Despite attempts to automate these controls as much as possible to avoid the need for human intervention, some controls invariably require interaction with users. Having users enter passwords, one of the most common types of controls, is, in fact, the most common authentication-related user task. Provided that the interaction sequence for password entry is reasonably simple and intuitive, users can accomplish this task rapidly and easily. But restrictions with respect to passwords that users can select based on certain settings or parameters—password length, age, combinations of characters that are allow, and more—are another entirely different matter. This paper explores the nature of these restrictions with the goal of weighing the costs versus benefits of each and also reviews research studies to determine whether empirical support for widely held preconceptions concerning the value of certain restrictions are in fact true. If not, information security may in effect be torturing users—forcing them to engage in actions that are difficult for humans to perform—in the name of security, even though these actions are of little or no benefit from a security perspective.

The "Straw Man"—Benchmarks for Passwords

A good starting point in examining the issue whether widely prescribed and used password settings are effective from a security perspective is to look at commonly used benchmarks for passwords. One of the most widely used password benchmarks have been developed by the Center for Computer Security (www.cisecurity.com). For example, consider the following Windows XP Windows benchmarks published by this organization:

Minimum password age - 1 day
Maximum password age - 90 days
Minimum password length: 8 characters
Password complexity - Enabled
Password history - 24
Store passwords using reversible encryption – Disabled

The first setting affects how long a user must keep the current password before the user is allowed to change it. The major reason for recommending a value other of 0 is to prevent users from changing their passwords when the are required to do so, then changing their passwords right back to the ones they had previously. The second setting requires users to change their passwords a minimum of once every 90 days. The third requires users to have at least eight characters in their passwords. In a Windows system that uses English, password complexity requires that a password contain at least three of the four following types of characters: an uppercase English alphabet character, a lowercase English alphabet character, a number and a special character such as % or &. Finally, reversible encryption means encryption based on the Data Encryption Standard (DES) for which encryption is incredibly easy to break. Disabling reversible encryption is thus critical from a security point of view (although disabling it may break compatibility with older Windows systems and applications). Each of these settings except for minimum password age and reversible encryption, neither of which has been the focus of any usability research, will now be analyzed.

Password Age

Research shows that frequent password changes are not good from a memorability standpoint. Bunting found that "proactive interference" from older passwords creates difficulty for users trying to remember their current passwords (BUNT06). When users feel that they cannot remember their passwords, they write them down, thereby often violating their organization's security policy. A survey of 3,050 Web users performed by Rainbow Technologies discovered that 55 percent of those surveyed confessed to writing down at least one password (RAIN03). Eight percent of survey respondents indicated that they wrote down *every* password that they had. A subsequent survey showed that 50 percent of users surveyed reported that they had written down at least one password, 10 percent reported that they *always* wrote their passwords down, and approximately 50 percent revealed that they frequently needed to have their passwords reset because they forgot them. The point here is that requiring users to change passwords frequently (e.g., once every 30 days, as is often required by banks) causes proactive interference, a form of memory interference, that results in failure to remember passwords. Users then turn to prohibited procedures, such as writing down passwords, thereby compounding the problem.

Another line of evidence concerning password age is less direct, but nevertheless very applicable. Today's password cracking tools (e.g., Cain and Able) are *incredibly* fast, so fast that the attempted cracking rate of Windows password cache password files (which are .PWL Files) on a Pentium 100 is 1,000,000 passwords per second (LOCK09). In other words, 1,000,000 candidate passwords can be compared to entries in .PWL files every second to determine if any

candidate password matches any entry in .PWL files. A typical recovery rates for ZIP or ARJ passwords on a Pentium 100 is 10,000,000 passwords per second on a fast or dual Processor PC (LOCK09). Perhaps most astounding is the fact that Distributed.net's Project Bovine RC5-64 computer can try 76.1 Billion passwords per second (LOCK09)! Given the speed with which passwords can now be cracked and given that someone (such as an attacker) who cracks a password is likely to use it right away to verify that it is valid, the difference between a password ago of 30 and 90 days, or even between 15 and 120 days, is now really quite inconsequential.

Minimum Password Length

A very short password, e.g., five characters in length, is an easy target for password crackers. But given the incredible speed of brute force password cracking, a password that is nine characters long is functionally no stronger than one that is eight characters long. Although the time difference depends on the amount of memory and processor speed on the computer on which a password cracking program runs, the time difference to crack a password consisting of one additional character is likely to be in seconds. The same applies to comparing a ten character long password to a nine character long one.

There is, however, a huge exception to the rule that increased password length does not make that much difference as far as time needed to crack passwords using modern password cracking tools. They tools, as good as they are, do not even attempt to crack Windows passwords that are at least 15 characters long. So a user who selects a horrible password such as "AAAAAAAAAAAAAAA" would at least survive even the most proficient password cracking tool's attempts to crack that password.

Password Filtering/Password Complexity

Windows XP's password complexity setting is more properly known as a "password filtering" setting. Password filters restrict the choice of characters that can be used in a password in an attempt to reduce the problem of users selecting passwords that are otherwise too easy to crack. Filtering rules usually impose restrictions on user-generated passwords, such as the previously described restrictions that the Windows XP's password complexity setting imposes.

To test the notion that password filters help passwords resist cracking attempts, Vu, Proctor, Bhargay-Spanzel, Tai, Cook and Schultz conducted an experiment in which seven password restrictions were imposed upon users who had to create passwords for their accounts (VU2007). The restrictions were that the password must:

- Be at least 6 characters long
- Contain at least one uppercase letter
- Contain at least one lowercase letter
- Contain at least one digit
- Contain a special character (e.g., ! or #)
- Be unique from the passwords generated for the other accounts
- Not contain the user's username or any variant of it

Users had to choose and remember passwords for 1, 3 or 5 accounts. The lc5 password cracking tool was used to attempt to crack all passwords for a total of four hours. Significantly fewer of the passwords from the 5-accounts group were cracked than for the 3-accounts group (40 percent versus 60 percent, respectively), but there the difference between these groups and the 1-account group was not statistically significant. There was no significant difference between groups in terms of the time needed to create each password and the login time. Forgetting was significantly highest for the 5-accounts group, e.g., 69 percent of the 5-accounts group was unable to recall the password for at least one of the five accounts, in contrast to 19 percent for the 3-accounts group and 15 percent for the 1-account group. Furthermore, passwords that users created had to satisfy *seven* password criteria, yet about half of these passwords were cracked within four hours

The results of this study have important implications for password settings, one of the most fundamental of which is that proactive password restrictions do not necessarily result in more crack-resistant passwords. The fact that such a high percentage of passwords were cracked by a password cracker (lc5), one that is by today's standards not all that powerful,l is additional support for this conclusion. Furthermore, having to remember more passwords that have been created under restrictions resulted in greater forgetting. The cost-benefit ratio of password filtering is thus questionable.

A good way to generate a password that fulfills complexity restrictions, but it potentially easier to remember is to create a passphrase. For example, the first characters in each word in "These are the times that try men's souls" can be used to create a password, "Tattttms." Do passphrases improve memory when filtering rules are used? Vu, Proctor, Bhargav-Spantzel, Tai, Cook, and Schultz undertook a study in which one group of participants had to create a passphrase that conformed to complexity restrictions (VU07). Another group had to create passphrases under the same restrictions, but also had to insert one digit and one special character into the passphrase. Results indicated that creating passphrases yielded more crack-resistant passwords *only* when users were also told to embed a digit and special character into the passphrase. Embedding a digit and special character also resulted in less ability to remember passwords during both short-term and long-term recall. Embedding digits and special characters resulted in significantly more time needed to generate and recall passwords and almost twice as many errors before they could recall the password. These results suggest that the widely held assumption that requiring users to create passphrases to improve both resistance to cracking and password memorability is more myth than fact**.**

Other Security Methods

Users also have numerous problems with other security methods that many information security professionals think are perfectly fine. For example, Proctor, Lien, Salvendy and Schultz conducted research on usability considerations in third-party authentication methods, methods that require something besides passwords during the authentication process (PROC00). These researchers conducted task analyses, breaking down users' tasks into individual, sequential steps to evaluate the number of task required in biometric-, smart card- and password-based authentication. In general, the greater number of steps needed to complete a task, the more difficult the task is for users—more time is likely to be required, and the number of errors is likely to increase. Proctor et al. discovered that in comparison to password authentication,

biometric devices necessitated 10 additional task steps. Compared to password-based authentication, smart cards required 14 additional task steps.

Results suggest an explanation concerning why third-party authentication methods have not gained in popularity as much as security needs would appear to mandate. Having to perform numerous additional steps in third-party authentication presents a significant usability hurdle to users, one that in all likelihood produces a great deal of user frustration and ultimately resistance to this type of authentication. In addition, certain steps identified in the task analyses were much more likely to result in user errors than others. For instance, inserting a smart card correctly into the card reader necessitated a series of steps that required exact orientation and manipulation of the smart card so that it could be put directly into the reader. Failure to orient and manipulate the smart card precisely resulted in errors on users' part.

User Resistance to Security Measures

Tasks and systems that have poor usability design cause users to resist them (e.g., Al-Ghatani & King, 1999). Resistance can manifest itself if numerous ways—negative statements, hostile behaviors, passiveness, failure to pay attention, circumventing security controls altogether, and in numerous other ways. Minimizing or eliminating altogether user resistance by considering the impact of human usability design should be a major part of the security controls selection process, but it is generally not. Instead, too often information security professionals develop a negative attitude towards the user community and then prescribe more security awareness and training for users as the solution to the problem. Unfortunately, *"user resistance to security" is too often in reality "user resistance to user-unfriendly security tasks!"*

A Realistic Assessment of Password-related Risks

Finally, it is important to consider the threats associated with password-related risks. Many information security professionals believe that password cracking tools lead to the greatest password-related risks. Although this used to be the case, most current attack methods do not involve password cracking, because it is not all that efficient— it almost always entails brute force password attacks—and also because gaining access to a password file requires superuser privileges, something that is not always easy to do if one is not a system administrator. Writing down passwords on slips of paper occurs even less than does password cracking nowadays.

Currently, keystroke and tty sniffers are the major threat vectors for password-related risk. Attackers perform reconnaissance activity that includes discovering individuals who frequently send email to each other and then craft special messages that appear to come from someone with whom one user frequently exchanges email messages. These messages either contain malware embedded within an Abode Reader attachment or a URL, which if clicked causes the browser to be redirected to a malicious Web site. Ultimately, perpetrators take control of targeted machines and then plant keystroke or tty sniffers to capture passwords and other credentials such as banking PIN numbers (SCHU09). By now it should be apparent that the quality of passwords as well as other password characteristics and rules (such as forbidding users to write down their passwords) make little difference in terms of the likelihood of success with today's password attacks. Why then do we fight such well-meaning, but ill-advised battles with users over

password settings such as the length, age, and complexity of passwords, or whether or not passwords can be written down?

Conclusion

In many ways, we torture users. Many of our beliefs and practices concerning passwords (and also other forms of authentication) clash with empirical research results. We have in reality invented our own "folklore" and then somehow labeled it "best practices." It is also extremely unlikely that many commonly used password policy settings produce anything close to a favorable cost-to-benefit ratio when the lost productivity of users who have to enter one password after another to satisfy password restrictions and also call the help desk when they cannot remember their difficult-to-crack, but also difficult-to-remember passwords. So why do we not instead switch to the use of one-time passwords, passwords that users do not have to create according to often difficult restrictions and that if captured during a login attempt, do attackers no good whatsoever?

References

ALGH99 Al-Ghatani, S. S., & King, M. (1999). Attitudes, satisfaction and usage: Factors contributing to each in the acceptance of information technology. *Behaviour & Information Technology*, 18, pp. 277-297.

BUNT06 Bunting, M (2006). Proactive interference and item similarity in working memory. *Journal of Experimental Psychology: Learning, Memory & Cognition*, 32(2), pp. 83-96.

LOCK09 Lockdown.co.uk (2009). Password Recovery Speeds: How long will your password stand up? July 10, 2009 from http://passwordresearch.com/stats/statindex.html

PROC02 Proctor, R. W., Lien, M. C., Vu, K.-P. L., Schultz, E. E., & Salvendy, G. (2002). Improving computer security for authentication of users: Influence of proactive password restrictions. *Behavior Research Methods, Instruments, & Computers*, 34, pp 163-169.

PROC00 Proctor, R.W., Lien, M., Salvendy, G. & Schultz, E.E. (2000). A task analysis of usability in third-party authentication. *Information Security Bulletin*, 5 (3), pp. 49 – 56.

RAIN03 Rainbow Technologies, 2003. Password survey results (June 2003). Retrieved November 14, 2005, from /http://mktg.rainbow.com/mk/get/pwsurvey03S.

SCHU09 Schultz, E.E. (2009). The new intrusion detection. Presentation at the SoCal Security Forum, Long Beach, CA, October 29, 2009.

VU07 Vu, K.-P. L., Proctor, R. W., Bhargav-Spanzel, A., Tai, B.-L., Cook, J., & Schultz, E. E. (2007). Improving password security and memorability to protect personal and organizational information. *International Journal of Human-Computer Studies*, 65, pp 744-757.