Reward Smarter Password Choices

by M. E. Kabay, PhD, CISSP
Associate Professor, Information Assurance
Norwich University, Northfield VT

Reader Andreas Englisch from Munich wrote to me with some interesting suggestions about improving password management. With his kind permission, here are some of his comments (by the way, readers are welcome to write to me in German and French as well as English; and with the help of my translation program I can also manage to muddle through Spanish, Italian and Portuguese).

* * *

If you force passwords to expire after a fixed interval, people tend to define passwords containing a number. When their passwords expire, they simply increase that number by one; e.g., password1, password2, password3, etc.

Most systems prevent reuse of a password for a set period; for example, “The last 15 passwords are saved and may not be re-used.” This password-numbering habit prevents a new password from being rejected by the system, but if anybody gets hold of such a password by shoulder surfing, dumpster diving, finding the sticky-note under the keyboard and so on, it is not going to be very difficult to find the next password in the series.

Moreover, I do not like password expiry by fixed intervals from another perspective: It treats all passwords the same, no matter how “good” (i.e. complex) they are. But would it not be better to set the password expiry interval as a function of password complexity? For example, if I use a really complex string of alphanumerical characters, I would have to change my password much less frequently than if I chose a more guessable password. This strategy would encourage people to use and remember better passwords than they currently use; in addition, they might stop using the same-old-password-1, same-old-password-2 approach to new passwords. Using this kind of approach, perhaps a bad password would have to be replaced after 30 days but a good one after 90 days and a really tough one after 180 days.

* * *

Dear Herr Englisch,

Your analysis of passwords with numbers is correct: any password with a number in it suggests that the next one will have similar characteristics. Therefore, password-checking algorithms should compare new passwords against old ones not only by a simple lookup list but also by using wild-card matching algorithms to detect static passwords that change in only one number or character or which shift all characters by the same increment of the sort sequence (e.g., “alpha1” becomes “bmqib2” and then later “cnnjc3”) (and no, these are not my password!).

However, I worry a bit about long password lifetimes. It seems to me that the primary reason for forcing password changes is that passwords can be compromised by inadequate security practices, as you yourself pointed out.
The cryptographic strength seems to be a lesser vulnerability. For example, an eight-character password with uppercase and lowercase letters and numbers available could take months to crack (see “Brute-Force Cracking Estimation” <http://www.mekabay.com/methodology/keyspace.xls>) depending on processing power, but a single instance of shoulder-surfing could compromise it within a day of its being changed.

So perhaps the reward for more complex passwords can be tempered by concern over possible compromise; 30-60-90 days, for example?

In any case, I appreciate your taking the time to write and thank you for letting me quote you. Danke sehr!

* * *

A Master’s degree in the management of information assurance in 18 months of online study from Norwich University – see <http://www3.norwich.edu/msia> for details.

M. E. Kabay, PhD, CISSP is Associate Professor in the Division of Business and Management at Norwich University in Northfield, VT. Mich can be reached by e-mail at <mailto:mkabay@norwich.edu>; Web site at <http://www.mekabay.com/index.htm>.

Copyright © 2005 M. E. Kabay. All rights reserved.

Permission is hereby granted to Network World to distribute this article at will, to post it without limit on any Web site, and to republish it in any way they see fit.