When I was a lad (OK, when I was a young systems engineer of 30) (which is 30 years ago), I was taught that if a user made several mistakes in entering her password, the system should lock her account until a system operator granted access again. The goal was to stop an attacker from guessing at a user’s password without limit.


“The second method recommended for mitigating guessing attacks is to configure OS and application password authentication mechanisms to limit the frequency of authentication attempts. Examples of how this can be accomplished include the following:

- Lock out a user account after a number of consecutive failed authentication attempts (often performed within a particular time period, such as the past hour). For example, after a user has failed to provide the correct password 50 times in a row, ignore all additional authentication attempts to the user account for 15 minutes. Locking out an account after only a few failed attempts has a significant impact on legitimate users and tends to cause them to choose simpler passwords or store their passwords insecurely, thus weakening security.

- Have a fixed or exponentially increasing delay after each failed authentication attempt. After the first failure, for example, there could be a five-second delay; after the second failure, a 10-second delay; after the third failure, a 20-second delay, and so on.”

I really like the second method, but I have taught students for years that locking users out of a system after a sequence of bad passwords is a policy that confers enormous power on an attacker. Armed with a list of userIDs, an attacker can simply enter a bogus password (e.g., “a”) repeatedly into logons for every userID – including perhaps that of root users, if they are subject to the same rule – and shut down access to the entire system. If operator intervention is necessary to reset the passwords for access, this denial of service can be a nightmare.


“Some systems react to online attacks by a simple rule that locks the account after a certain number of failed attempts. This rule may have been borrowed from a similar rule with ATM cards. The rule actually makes sense in the context of ATMs, with two-factor authentication based on possession of the card and knowledge of the PIN. However, in a
password-only scheme, the ‘three strikes and out’ rule can lead to denial of service to legitimate users. An attacker can easily lock up many accounts by entering three wrong passwords repeatedly. A more graceful rule would slow down the rate at which password guessing can be attempted, so that a legitimate user may be perceptibly slowed down in authentication but not denied. For example, locking an account for a couple of minutes after three bad passwords suffices to make brute-force guesswork impractical.

In addition, intrusion-detection systems can be configured to alert system administrators immediately upon repeated entry of bad passwords. Human beings then can intervene to determine the cause of the bad passwords—user error or malfeasance.”

Here’s what I wrote in my comments to the NIST authors:

“Slow down user logins by inserting a fixed but humanly acceptable delay between authentication attempts. For example, allow successive logins to occur no more quickly than once every five seconds. The user will barely notice the delay, but any automated password-guessing program will be slowed into ineffectiveness. More important, this policy deprives an attacker the power to create a denial-of-service attack simply by trying the same wrong password a number of times in succession.”

I also suggested that the authors add this bullet:

“Ensure that repeated authentication failures activate an alarm for human operators. System operators may be able to learn details of who is attacking their system by observing the failed attempts to impersonate authorized users; computerized data collection and human observation may serve as evidence in legal proceedings and as a basis for improving security measures.”

I hope that as an industry, we can move away from inactivation of accounts in response to bad passwords and to a more intelligent response.

* * *

NIST requests comments on draft SP 800-118 by May 29, 2009. Please submit comments by e-mail <mailto:800-118comments@nist.gov> with "Comments SP 800-118" in the subject line.

* * *

M. E. Kabay, PhD, CISSP-ISSMP <mailto:mekabay@gmail.com> specializes in security and operations management consulting services. CV online.<http://www.mekabay.com/cv/>

Copyright © 2009 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.