;login:

THE MAGAZINE OF USENIX & SAGE

April 2003 • volume 28 • number 2

inside:

USENIX & SAGE

The Advanced Computing Systems Association &
The System Administrators Guild

hacking for fun and profit

by Tina Darmohray

Tina Darmohray, contributing editor of *;login:*, is a computer security and networking consultant. She was a founding member of SAGE. She is currently a Director of USENIX.

<*tmd@usenix.org*>

As terms borrowed from classic American westerns, often inhabited by black-hatted villains and white-hatted heros, a "black hat" cracker describes someone who breaks into a computer system or network with malicious intent; a "white hat" is a cracker who identifies a security weakness in a computer system or network so that the system's owners can fix the breach before it is exploited. White-hat cracking is a hobby for some while others provide their services for a fee. The paid white-hat cracker may work as a consultant or be a permanent employee on a company's payroll.

Regardless of hat color, both kinds of crackers are trying to achieve the same goal: successfully breaking into networks and computers. When used by the white hats, these targeted break-in attempts are often part of an overall security plan and are referred to as penetration tests. While you never know what you're going to find when embarking on such a test, it turns out that what they are (and are not) good for is well understood by security professionals. I asked over a dozen such individuals when they'd recommend penetration tests for an organization and got a lot of feedback, with a few variations on the theme. So if you're wondering what penetration testing can do for you, read on.

First off, penetration testing is part of an overall security plan. Security testing, like many other pieces of a total security solution, is not going to provide security to your site all by itself. And, in fact, penetration testing is really a post-implementation verification tool rather than something like a firewall or VPN concentrator, which are actually implementing part of the security at a site. There's a lot of legwork to be done before any verification is begun. Appropriate use of a penetration test implies that the up-front work is complete and is ready to be tested. The professionals reiterate that penetration testing is not a replacement for careful security design and implementation and that these designs begin with a thorough risk assessment and management buy-in of the organization's IT security priorities.

Once the security goals have been hammered out and a solution is in place, it's time to haul out the white hats and verify you've hit your target. Penetration testing is recommended to verify a site's initial security deployment and as a follow-up whenever new systems are deployed or existing systems are reloaded, upgraded, reconfigured, or patched, or when there are code changes to exposed services or applications. Thereafter, on a semiannual or annual basis, penetration tests are recommended to ensure that you're maintaining the security stance you've adopted and that your exposure profile has not degraded. Major vulnerabilities that lead to remote privileged access come out all of the time, so even though you passed the last test with flying colors, it helps to have one done at regular intervals. One frequent recommendation is to consider hiring someone else to do some of these follow-up tests, just to get a fresh set of eyes looking at your site.

Whether a penetration test is hired out or performed in-house, the tools and methodology used can determine the true value of the test results. Simply pointing a scanner at a site is not penetration testing; there must be a diligent effort to look at the site from several different angles, using a myriad of tools and some hand inspection of custom code and components. Better testers continue until they have found at least one vulnerability. The best testers carry out a test plan that includes a list of planned tests, so that they test much more than just the firewall or public servers, but look for other ways into the site.

Penetration tests are the technological equivalent of having the manager of the local bank, before going home, walking about to ensure that all of the windows, doors, and safes are shut.

Aside from straightforward verification, penetration tests appear to be used for a variety of other, perhaps more political, reasons. Some organizations use them as "feel good" reassurance which they provide to third parties to prove a certain level of security is in place. Sadly, sometimes it's necessary to use them to assist in in-house battles: "Hey! You can't put that application on the Web server; it's a huge vulnerability!" "Well, it's too late, we've gone live with it, and we won't shut it off unless you prove there's a problem." You know what happens next . . .

Prevailing practice indicates, however, that penetration tests have their place in the standard security suite, with the proviso that most of the security risk is still from authorized users. This is the old branch bank scenario. Penetration tests are the technological equivalent of having the manager of the local bank, before going home, walking about to ensure that all of the windows, doors, and safes are shut. This will not prevent the assistant manager from giving the key to an in-law who then breaks in on Sunday night. Similarly, a penetration test will not stop the technological equivalent of the assisted in-law, but the embarrassment factor and concomitant loss of business to a bank that loses money because the window was open is much greater than the errant in-law loss. That is true even if the in-law losses are larger than the loss as a function of the open window. It is the appearance that the basics weren't done right that is most damaging.

One last thing to keep in mind with regard to penetration testing: If it isn't successful, that doesn't mean you're safe, just that the right thing wasn't tried. It's the old problem of trying to prove the negative, and whether you do it yourself or hire it out, there is only so much time.