[in]security

by Gary McGraw

Is Application Security Training Worth the Money? : : :

Software security—sometimes called application security by the myopic—is catching on. That's good because we can certainly use less broken software in the world. But it's bad because there aren't enough knowledgeable people to build secure software. You see, the people who build software know next to nothing about security. It's no wonder they keep cranking out the security holes. One partial solution is to train your developers.

The problem is that everyone and their brother seem to be hanging up a shingle to teach about software security. Asking a potential instructor the right questions will determine whether you end up being shafted or actually affect the way your developers build software.

BEYOND FEATURES AND BUGS

Watch out for curricula built around security features alone. Although cryptography, a prime example of a security feature, is interesting to developers, you can't just liberally apply it to solve the software security problem. Developers are trained from birth to think about features and functions. They'll think (incorrectly) that a course on security features is just what the doctor ordered. But it doesn't work that way.

> Look for training that focuses on identifying and expunging problems in the software itself.

It's better to teach developers about software security touchpoints such as code review with a source code analysis tool and architectural risk analysis than it is to teach them about the latest glittery security software.

An even more pervasive problem in software security training is what I like to call the "bug parade." Software security problems come in two flavors: implementation bugs (such as a buffer overflow) and architectural flaws (such as incorrect remote method sharing). If your training course focuses only on bugs, you'll overlook at least half the problem.

Make sure any course you consider gets past the buffer overflow on line 42 and delves into security design principles, software architecture, and how to carry out various methods of software analysis.

AVOID NETWORK SECURITY INSTRUCTORS

The biggest problem of all happens when network security experts try to teach software security courses. Developers are a hard lot to please, and only a software veteran who has built real code will get and hold their attention. Network security people usually don't have the software chops. As traditional sources of network security training such as the SANS Institute expand to take on software security, there's a real danger that software will get the short shrift. Make sure any course you choose has as much software depth as it has security depth.

One good way to avoid the "security guy-only" problem is to stay away from "application security" or "Web application security" courses. The term "application security" unnecessarily limits the purview of software security. Sure, applications have security problems, with Web-based applications leading the pack. But if you step back a moment, you'll see we have a much bigger problem than simply errant Web applications. Ask yourself, what do wireless devices, cell phones, PDAs, browsers, OSs, routers, servers, PCs, PKI systems, and firewalls have in common? The answer is software. Real attackers go after bad software no matter where it lives. Focusing on "application" code ignores the bigger picture.

Smart architects will look for training that revolves around software security and focuses on an inside-out approach in which the process of designing, building, and testing software for security identifies and expunges problems in the software itself.

If the course you choose is 1) taught by software people, 2) focuses on design flaws as well as bugs, and 3) covers software security touchpoints, you'll find it worth the money. Otherwise you might as well spend it on beer.

Gary McGraw is CTO of Cigital, a software quality management consultancy. He is co-author of *Exploiting Software* (Addison-Wesley, 2004), *Building Secure Software* (Addison-Wesley, 2001), and *Java Security* (Wiley, 1996). His most recent book is *Software Security* (Addison-Wesley, 2006). Reach him at gem@cigital.com.