November 2003

SC Infosec Opinionwire

Network & Security Technologies

Security Awareness for Software Developers:
"An Ounce of Prevention or a Pound of Cure"

by Adam Lipson

"An Ounce of Prevention" or "A Pound of Cure?"

Successful attacks against large enterprises by new viruses, hackers, industrial spies and cyber-terrorists are a daily occurrence. Most could have been avoided if those writing and maintaining software practiced fundamental principles of secure software development.

Correcting flaws identified during software security audits is expensive and time consuming. Worse, vast resources are spent on containing and recovering from exploits. Fortunately, providing development staff with the knowledge and tools to avoid many of these pitfalls is easy and inexpensive.

The choice is simple. Spend large sums repairing damage caused by faulty software already deployed. Or give developers a set of inexpensive tools to build better and more secure applications in the first place.

This "ounce of prevention" should be focused on the following objectives:

- Increase application security awareness of programmers, designers, software development project managers and security professionals. Sponsor training that promotes your organization's security policy, educates your developers and specifies where development fits into the overall security policy.
- Provide information about typical attacks used against application software and the guiding principles to successfully mitigate the risk. Demonstrate how common exploits such as cross-site scripting, cookie poisoning or buffer overflow vulnerabilities impact applications. Show how these exploits could have been avoided through integration of proper security principles into the development process.
- Empower project managers and security professionals to evaluate the security built into applications--whether developed in-house, purchased or outsourced. Developing the security testing procedure is NOT something that should be performed after an application is written. Proper security tests should be developed prior to writing code.

- Enable programmers and designers to prove the security of their products against high-risk vulnerabilities and exploits through proper coding and testing techniques. These may include integrating routine code reviews into your development culture.

Good security is proactive--not reactive. Train your development staff today and you will save money tomorrow. Training will promote security and improve protection of your corporate assets while building lasting bridges between your development and security organizations. It may also result in a few less sleepless nights fighting fires.

Adam Lipson is president and CEO at Network and Security Technologies (www.netsectech.com), a provider of digital security consulting solutions.
