



Software Assurance Forum for Excellence in Code

*Security Engineering Training:
Building the Foundation for Software
Security Success*

March 2012



The Software Assurance Forum for Excellence in Code (SAFECode) is a global, industry-led effort to identify and promote best practices for developing and delivering more secure and reliable software, hardware and services





Fundamental to our success:

- The people who are designing, developing, testing and delivering software must understand the fundamentals of secure software engineering that lead to secure practices and products





Today's reality:

- Every SAFECode member software assurance program is supported by an internally developed training program
- It is the only way to build specialized skills and knowledge to support their own organization's unique development environment
- This does not mean there is not value in university programs and professional certification/external training programs



Content should create an understanding of basic security such as:

- Secure Design Principles
- Secure Coding Principles
- The Most Common Errors that Lead to Security Vulnerabilities
- Threat Modeling
- How to Find / Test Security-related Issues in Code
- How to Fix Security Issues etc.
- Security in the company's Software Development Lifecycle

A conceptual understanding of security issues should include buffer overflows, data validation, SQL injection, cross site scripting, format string vulnerabilities and use of unsafe functions or behaviors, etc.



Target a Broad Internal Audience

- Spread awareness of software security
 - Develop “security-aware” culture & mindset
- Develop advocates in Engineering
- Target all who touch products !!!



SAFECode member companies believe that the basics of security engineering need to be understood by everyone involved with software development including Product Managers, Product Architects/ Designers, Program/ Project Managers, Development Engineers and Quality Assurance (QA) Engineers.



- Basic understanding of the current threat environment and the important role of secure development practices in attack prevention
- Overview of business risks/rationale, internal standards and policies, basics of secure coding and testing
- Overview of a company's approach to security in the development lifecycle

Intended for all employees involved in product development and management.



- Language & OS specific techniques to prevent and fix software vulnerabilities
- Secure design principles
- Secure testing methodologies
- Secure coding techniques
- Find and fix security flaws against common definitions (e.g. CWE, CVE)
- Threat modeling techniques



Intended for all employees involved in product development and testing



- Role-based
- Directly tied to job functions
 - Security tools (e.g., specific static code analysis tool)
 - Specific technology implementations and their associated security concerns
 - Cryptography



Intended for engineers seeking to improve their security knowledge in specialized areas.



- While some of the skills / knowledge required are static; others need constant education
- It is important to develop an understanding of the ever-changing, dynamic threat
- Many SAFECode members supplement their internal training programs with informal approaches to keep product teams updated on security developments
 - podcasts, newsletters, in-house conferences, webinars, guest speakers, etc.



- Must be relevant to work at hand
- Sensitive to corporate culture
- Mandated vs. professional development
- Must reward people; develop incentives
- Keep direct link between content and performance





Time is most the precious commodity

- Customized and adaptable
- Applying the techniques lead to less time to fix errors
- Tailor to product development cycles





- Computer Based Training vs. Instructor Led Training debate
 - Depends on: company's unique attributes, size, culture, distribution
 - SAFECODE members: Adopt Hybrid

Computer Based	Instructor Led
Flexible Schedule	Get direct answers
Cost Effective	May accommodate labs
Suited for global orgs	In-house mentoring
Can intermix COTS content	Build peer resources



- Tie training measurement to corporate goals (and individual performance)
- Direct and in-direct measurements
 - How many team members have been training?
 - How is training being received by learners?
 - Has it affected quality of output - i.e. teams with more trained engineers produce code with fewer vulnerabilities than those teams with less



- Though not a replacement for formal education on secure development principles and practices at the university level, in-house training is essential to success
 - Industry must drive its own improvements; can't afford to wait for someone else to train workforce
 - This is not without challenges
 - SAFECode working to help address some of these challenges



For More:



Security Engineering Training: A Framework for Corporate Training Programs on the Principles of Secure Software Development

- http://www.safecode.org/publications/SAFECode_Training0409.pdf

All SAFECode Papers available at
www.safecode.org at no cost

For more:

Stacy Simpson
SAFECode Policy Director
stacy@safecode.org