

SC DEV 201: SECURE CODING IN JAVA EE

After taking this advanced class students will be able to develop secure Java Enterprise Edition (J2EE) applications. Students will learn to define and identify secure code, differentiate between secure coding methods, employ secure code in practice and design and judge effectiveness of secure coding practice.

The class focuses on learning by doing. Concepts are presented in short lecture-demonstration sessions, and then students are challenged in hands-on labs to make reasoned choices and implement secure code. Each heading in the detailed outline includes Hands-On labs with applications that must be modified in some way to make them secure. Students are required to execute various real world solutions including fixing broken applications, adding security functionality, replacing poorly written code, finding vulnerabilities and doing runtime testing.

All labs are executed in a real world, preconfigured development environment. Days 1 and 2 focus on web applications and day 3 focuses on core Java security. Students secure coding abilities will be materially sharpened after this class.

DURATION & INTENDED AUDIENCE

- Duration: 3 days
- Intended Audience:
 - Java EE Developers
 - Java EE Architects
 - Java Programmers

PRE-REQUISITE KNOWLEDGE

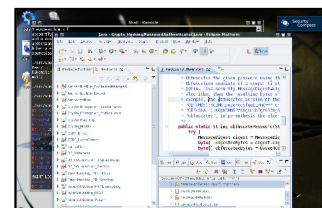
- Knowledge of common application security vulnerabilities (e.g. OWASP Top 10) mandatory
- Strong understanding of (J2EE) Java Enterprise Edition mandatory
- Understanding of Apache Struts and Spring Frameworks recommended
- Experience with Eclipse IDE recommended

LEARNING OBJECTIVES

- Understand what various defensive technologies do and how they work
- Gain hands-on experience in writing code to add security controls into vulnerable apps
- Evaluate alternatives for application security solutions
- Understand less publicized areas of application security (e.g. concurrency)

LEARNING ENVIRONMENT & TAKEAWAYS

- 4 GB bootable Backtrack Linux flash drive that students keep
- Eclipse, Tomcat, My SQL, and source code from over 30 different **real** open source applications
- Course book containing printouts of each slide along with detailed notes in paragraph form
- Hard and soft copy of a secure coding checklist featuring class topics



CONTACT INFO

- Phone: 1-888-777-2211 x 1
- Email: training@securitycompass.com

DETAILED OUTLINE

DAY 1: INPUT VALIDATION

Input Validation

- Attack Review
- Bug Hunt
- Regular Expressions
- Servlet Filters
- Struts Validation
- Output Encoding
- Anti-CSRF tokens
- Anti-SCRF CAPTCHA
- Prepared Statements
- Stored Procedures
- AOP Introduction
- AOP Input Validation

DAY 2: ACCESS CONTROL

Authentication and Session Management

- Attack Review
- Container-based authentication
- Client Certificates
- Acegi authentication
- Jasypt passwords
- Other considerations
- Anti-session Fixation Active Session Timeouts
- IP-Session Correlation Authorization

Authorization

- Authorization tiers
- Coarse & Fine-grained Auth
- Role Based Access
- Declarative Access Control
- Programming Access Control
- JAAS
- Spring Security Authorization

DAY 3: SECURE JAVA

Exception Handling and Logging

- Errors and Security
- Denial of Service
- Runtime vs. Checked
- Exception Handling Gaps
- Logging Frameworks
- Sensitive Data Logging
- Centralized Logging and Prevention of Attacks

Cryptography

- Secure Random
- Channel Security
- DB Encryption with Jasypt
- Java Crypto Packages
- PKI
- Cryptographic controls