Secure Java Coding (TT8200-J)

Developing Secure Standalone and Desktop Java Applications

This course is a hands-on, lab-intensive Java security, code-level training course that teaches you the best practices for designing, implementing, and deploying secure programs in Java. You will take an application from requirements through to implementation, analyzing and testing for software vulnerabilities. This course explores well beyond basic programming skills, teaching developers sound processes and practices to apply to the entire software development lifecycle. Perhaps just as significantly, you will learn about current, real examples that illustrate the potential consequences of not following these best practices.

Although this edition of the course is Java-specific, it may also be presented using .NET (TT8200-N) or other programming languages.

What You'll Learn

- Concepts and terminology behind defensive coding
- Threat Modeling as a tool in identifying software vulnerabilities based on realistic threats against assets
- Entire spectrum of threats and attacks that take place against software applications
- Threat Modeling to identify potential vulnerabilities in a real life case study
- Static code reviews and dynamic application testing for uncovering vulnerabilities in Java applications
- Vulnerabilities of the Java programming language and the JVM, and how to harden both
- Work with Java 2 platform security to gain an appreciation for what is protected and how
- Roles that Java Authentication and Authorization Service (JAAS) have in Java applications
- Use JAAS in conjunction with a Java application for both authentication and authorization
- Basics of Java Cryptography (JCA) and Encryption (JCE) and where they fit in the overall security picture
- Fundamentals of XML Digital Signature and XML Encryption

Who Needs to Attend

Application project stakeholders who wish to develop secure Java applications

Prerequisites

- Familiarity with Java and JEE is required
- Programming experience is highly recommended
- At least six months of Java and JEE working knowledge recommended
- You should have a working knowledge in the following topics or attend these courses as a prerequisite:

- Java 7 SE Programming for OO Experienced Developers (TT2100-J7)
- Introduction to Java 7 Programming for Non-OO Developers (TT2120-J7)
- Java Web Essentials for OO Developers (TT5140)

Follow-On Courses

- Securing JEE Web Services (TT8500-JEE)

Course Outline

1. Introduction: Misconceptions

- Security: The Complete Picture
- TJX: Anatomy of a Disaster?
- Causes of Data Breaches
- Heartland - Slipping Past PCI Compliance
- Target's Painful Christmas
- Meaning of Being Compliant
- Verizon's 2013 Data Breach Report

2. Foundation

- Security Concepts
  - Motivations: Costs and Standards
  - Open Web Application Security Project
  - Web Application Security Consortium
  - CERT Secure Coding Standards
  - Assets are the Targets
  - Security Activities Cost Resources
  - Threat Modeling
  - System/Trust Boundaries
- Principles of Information Security
  - Security Is a Lifecycle Issue
  - Minimize Attack Surface Area
  - Layers of Defense: Tenacious D
  - Compartmentalize
  - Consider All Application States
  - Do Not Trust the Untrusted
- Vulnerabilities
  - Unvalidated Input
  - Broken Access Control
  - Broken Authentication And Session Management
  - Cross Site Scripting (XSS) Flaws
  - Injection Flaws
  - Error Handling And Information Leakage
  - Insecure Storage
  - Insecure Management of Configuration
  - Direct Object Access
  - Spoofing and Redirects
- Understanding What's Important
  - Common Vulnerabilities and Exposures
  - OWASP Top Ten for 2013
  - CWE/SANS Top 25 Most Dangerous SW Errors
  - Monster Mitigations
  - Strength Training: Project
  - Teams/Developers
  - Strength Training: IT Organizations

3. Java Security

- Java Security Fundamentals
  - Perimeter Defenses
  - Java Security Architecture
  - JVM Defenses
  - Extending the defenses
- Cryptography Overview
  - Strong Encryption
  - Ciphers and algorithms
  - Message digests
  - Keys and key management
- Code Location-Based Security
  - Work with Java 2 Security
  - Byte Code verifier
  - Signing code
  - Trusted code
  - Java permission management
  - Extending Java permissions
- User-based J2SE Security
  - JAAS Authentication
  - Extending JAAS authentication
  - JAAS Authorization
- Java Network Security

- SSL Support
- HTTPS
- GSS
- SASL protocols

- Code Level Security Best Practices
  - What Java security provides for
  - Preventing remote hacking
  - Preventing accessing of restricted resources
  - Retaining credibility with Java code

4. Defending XML and Services

- Defending XML
  - XML Signature
  - XML Encryption
  - XML Attacks: Structure
  - XML Attacks: Injection
  - Safe XML Processing
- Defending Web Services
  - Web Service Security Exposures
  - When Transport-Level Alone is NOT Enough
  - Message-Level Security
  - WS-Security Roadmap
  - XWSS Provides Many Functions
  - Web Service Attacks
  - Web Service Appliance/Gateways

Labs

Hands- on Learning: As a programming class, this course provides **multiple challenges** labs for students to work through during the class. This workshop is about **50% hands-on lab and 50% lecture**. Throughout the course students will be led through a series of progressively advanced topics, where each topic consists of lecture, group discussion, comprehensive hands-on lab exercises, and lab review. Multiple detailed lab exercises are laced throughout the course, designed to reinforce fundamental skills and concepts learned in the lessons. At the end of each lesson, developers will be tested with a set of review questions to ensure that he/she has fully understands that topic.

Private Group Training

Course Code: 1147

Contact us for pricing

3 Day Course

Date created: 2/10/2016 11:25:02 PM