Java Secure Coding

for

Client-Server Applications

4-Day hands on Course

Course Syllabus

AppSec Labs Ltd.

info@appsec-labs.com | https://appsec-labs.com | 10 HaTa'as St., Kfar Saba 44641 Israel | T: +972-9-7485005 | F: +972-9-7730595

Java Secure Coding for Client-Server Applications 4-Day hands on Course

Course description

Secure programming is the best defense against hackers. This multilayered hands-on course will demonstrate live real time hacking methods, analyze the code deficiency that enabled the attack and most importantly, teach how to prevent such vulnerabilities by adopting secure coding best practices in order to bullet-proof your J2ee application.

The methodology of the cycle of knowledge is as follows: understand, identify, prevent. This methodology presents the student with analytical tools to keep a deeper understanding of coding vulnerabilities and implement security countermeasures in different areas of the software development lifecycle. The hands on labs will enable the student to get a firsthand experience of the hackers' world and what could be done to stop him. Using sound programming techniques and best practices shown in this course, you will be able to produce high-quality code that stands up to attack.

The course covers major security principles in the Java framework, programming vulnerabilities, and specific security issues in J2EE web applications.

Target audience

Members of the software development team:

- Java developers in J2EE based applications
- Designers & architects

Prerequisites

Before attending this course, students should be familiar with:

- Basic knowledge of the Java framework
- Apache/Tomcat, Databases (MySQL/Oracle) & SQL language

Course topics

Day 1:

Authentication

- Password storage
- Hashing
- Brute force
- Anti-automation
- User enumeration
- Authentication types
- Two-Factor Authentication
- Kerberos
- JAAS
- **Labs: Hands-On Authentication**

Authorization

- Client side authorization
- Direct URL Access
- UI-based security
- Parameter tampering
- Direct object reference
- File authorization
- ACL vs RBAC
- Java Security Manager
- JAAS authorization
- OAuth
- **Labs: Hands-On Authorization**

Input Validation

- Integer Overflows & Underflows
- OS Command Injection
- SQL Injection
- Parameterized queries
- Stored procedures
- XPATH Injection
- LDAP Injection
- Strong typing
- Blacklist vs. Whitelist validation
- Regular expressions (Regex)
- **Labs: Hands-On Input Validation**

Day 2:

File Handling

- Directory traversal
- Canonicalization
- Backdoors
- File extension handling
- Directory listing
- Isolation storage
- ACL
- **Labs: Hands-On File Handling**

Data Confidentiality & Integrity

- Introduction to Crypto
- Privacy Standards
- Insecure storage
- Symmetric encryption
- A-Symmetric encryption
- Insecure communication
- Secure traffic enforcement
- Hash functions
- Digital signatures
- Randomization
- **Labs: Hands-On Crypto**

Application Denial of Service Vulnerabilities

- Application / OS crash
- CPU starvation
- Memory starvation
- File system starvation
- Resource starvation
- Resource locking
- Triggering high network bandwidth
- User level DoS
- Exploiting a specific vulnerability to cause DoS
- **Labs: Hands-On DoS**

Day 3:

Code Protection

- Reverse engineering techniques
- Obfuscation
- Native compilers
- Jar protection - Signed jar, Sealed jar
- Digitally signed applets
- Secure object serialization
- **Labs: Hands-On Code Protection**

Error Handling

- Information disclosure
- Exceptions and stack trace
- Default error pages
- **Labs: Hands-On Error Handling**

Security Logging

- Logging technologies
- Events that should and should not be logged
- Integration with exception management
- **Labs: Hands-On Auditing**

Business Logic

- Logical attacks
- Flow bypassing
- Replay attacks
- Abuse of functionality
- Race conditions
- **Labs: Hands-On Business Logic**

Day 4:

Secure Design and Principles

- Implementing security as part of the SDLC
- Secure Design
- Common Security Principles
- Layered Security

Java SE Security

- What is Java SE?
- Platform Security
- Bytecode Verification
- Secure Class Loading
- Crypto APIs
- Cryptographic Service Providers
- Authentication APIs
- Access Control
- Secure Communication APIs
- Public Key Infrastructure (PKI)

Advanced Topics in Java Security

- Java Servlet Filter
- JSSE (Java Secure Socket Extension)
- JAX –RS: securing RESTful Web-Services
- Security Code Review: how to find security holes in your code
- Security Tools and Scanners
- Penetration Testing: do it yourself

Spring Security

- Security Concepts
- Authentication Types
- Authorization Manager
- Security Interceptor
- Roles and Expression-Based Access Control
- Spring Security Configuration
- Security Filters
- Spring Security Crypto Module