Secure Web Application Development Seminar (Language Neutral Edition)

Course Summary

Description

This course is an intense web application security training workshop/seminar essential for web developers who need to produce secure web applications, integrating security measures into the development process from requirements to deployment and maintenance. 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, students learn about current, real examples that illustrate the potential consequences of not following these best practices. This course is short on theory and long on application, providing students with in-depth, code-level demonstrations and walkthroughs. This course is taught in a language-neutral fashion, with demonstrations from several languages to illustrate patterns and techniques.

In this course, students are shown best practices for defensively coding web applications, including XML processing and web services. Demonstrations repeatedly attack and then defend various assets associated with a fully-functional web application. This approach illustrates the mechanics of how to secure web applications in the most practical of terms.

Objectives
At the end of this course, students will be able to:

- Understand potential sources for untrusted data
- Understand the consequences for not properly handling untrusted data such as denial of service, cross-site scripting, and injections
- Be able to test web applications with various attack techniques to determine the existence of and effectiveness of layered defenses
- Prevent and defend the many potential vulnerabilities associated with untrusted data
- Understand the vulnerabilities of associated with authentication and authorization
- Be able to detect, attack, and implement defenses for authentication and authorization functionality and services
- Understand the dangers and mechanisms behind Cross-Site Scripting (XSS) and Injection attacks
- Be able to detect, attack, and implement defenses against XSS and Injection attacks
- Understand the concepts and terminology behind defensive, secure, coding
- Understand the use of Threat Modeling as a tool in identifying software vulnerabilities based on realistic threats against meaningful assets
- Perform both static code reviews and dynamic application testing to uncover vulnerabilities in web applications
- Design and develop strong, robust authentication and authorization implementations
- Understand the fundamentals of XML Digital Signature and XML Encryption as well as how they are used within the web services arena
- Be able to detect, attack, and implement defenses for XML-based services and functionality
- Understand techniques and measures that can used to harden web and application servers as well as other components in your infrastructure
- Understand and implement the processes and measures associated with the Secure Software Development (SSD)

Secure Web Application Development Seminar (Language Neutral Edition)

Course Summary (cont'd)

- Acquire the skills, tools, and best practices for design and code reviews as well as testing initiatives
- Understand the basics of security testing and planning
- Work through a comprehensive testing plan for recognized vulnerabilities and weaknesses

Topics

- Foundation
- Top Security Vulnerabilities
- Best Practices
- Defending XML Processing
- Secure Software Development (SSD)
- Security Testing
- Appendix: Security Design Patterns

Audience

This is an intermediate-level course designed for application project stakeholders who wish to get up and running on developing well defended web applications.

Prerequisites

Familiarity with a programming language (such as Java, .Net or C++) is required, and real world programming experience is highly recommended.

Duration

Two days

Secure Web Application Development Seminar (Language Neutral Edition)
Course Outline

I. Foundation

A. Misconceptions
1. Thriving Industry of Identify Theft
2. Dishonor Roll of Data Breaches
3. TJX: Anatomy of a Disaster
4. Heartland: What? Again?

B. Security Concepts
1. Terminology and Players
2. Assets, Threats, and Attacks
3. OWASP
4. CWE/SANS Top 25 Programming Errors
5. Categories
6. What they mean to your web applications

C. Defensive Coding Principles

D. Reality
1. Recent, Relevant Incidents
2. Find Security Defects In Web Application

II. Top Security Vulnerabilities

A. Unvalidated Input
1. Description With Working Example
2. Defenses
3. Identifying Trust Boundaries
4. Qualifying Untrusted Data
5. Implementing An Effect, Layered Defense
6. Designing An Appropriate Response
7. Testing Defenses And Responses

B. Overview Of Regular Expressions
1. Description With Working Example

C. Broken Access Control
1. Description With Working Example
2. Defenses
3. Authorization Security Overview
4. Defending Special Privileges Such As Administrative Functions
5. Application Authorization Best Practices

D. Broken Authentication And Session Management
1. Description With Working Example
2. Defenses
3. Multi-Layered Defenses Of Authentication Services
4. Password Management Strategies
5. Password Handling With Hashing
6. Mitigating Password Caching
7. Testing Defenses And Responses For Weaknesses
8. Alternative Authentication Mechanisms
9. Best Practices For Session Management
10. Defending Session Hijacking Attacks
11. Best Practices For Single Sign-On (SSO)

E. Cross Site Scripting (XSS) Flaws
1. Description With Working Example
2. Defenses
3. Character Encoding Complications
4. Blacklisting
5. Whitelisting
6. HTML/XML Entity Encoding
7. Trust Boundary Definition
8. Implementing An Effective Layered Defense
9. Designing An Appropriate Response
10. Cross-Site Request Forgeries (CSRF)
11. Understanding CSRF
12. Defending Against CSRF
13. Output Encoding – Why
14. Output Encoding – How
15. Output Encoding – Best Practices

F. Injection Flaws
1. Description With Working Example
2. Defenses
3. Qualifying Untrusted Data
4. Hibernate Best Practices
5. XML Best Practices
6. Third Party API's
7. Implementing An Effective Layered Defense
8. Designing An Appropriate Response

G. Error Handling And Information Leakage
1. Description With Working Example
2. Defenses
3. Web Application Exception Handling Framework
4. Error Response Best Practices
5. Error, Auditing, And Logging Content Management
6. Error, Auditing, And Logging Service Management
7. Best Practices For Supporting Web Attack Forensics

H. Insecure Storage
1. Description With Working Example

Secure Web Application Development Seminar (Language Neutral Edition)
Course Outline (cont'd)

2. Defenses
3. Data Leakage
4. Risk Minimization
5. Cryptography Overview
6. Data Encryption
7. Partial/Complete
8. Property/Deployment/Configuration Files

I. Insecure Management Of Configuration
1. Description With Working Example
2. Defenses
3. System Hardening
4. Server Configuration "Gotchas!"
5. Hardening Software Installation

J. Direct Object Access
1. Description With Working Example
2. Defenses
3. XML/DTD/Schema/XSLT Best Practices

K. Spoofing
1. Description With Working Example
2. Defenses
3. Protecting Your Clients
4. Defending Against Cross Site Request Forgeries
5. Phishing Defenses

III. Best Practices

A. Best Practices and Principles
1. Security Is A Lifecycle Issue
2. Minimize Attack Surface
3. Manage Resources
4. Application States
5. Compartmentalize
6. Defense In Depth - Layered Defense
7. Consider All Application States
8. Not Trusting The Untrusted
9. Security Defect Mitigation
10. Leverage Experience

IV. Defending XML Processing

A. Defending XML
1. Understanding Common Attacks And How To Defend
2. Operating In Safe Mode
3. Using Standards-Based Security
4. XML-Aware Security Infrastructure

B. Defending Web Services
1. Security Exposures
2. Transport-Level Security
3. Message-Level Security
4. WS-Security
5. Attacks And Defenses

C. Defending Ajax
1. Ajax Security Exposures
2. Attack Surface Changes
3. Injection Threats And Concerns
4. Effective Defenses And Practices

V. Secure Software Development (SSD)

A. SSD Process Overview
1. CLASP Defined
2. CLASP Applied

B. Asset, Boundary, and Vulnerability Identification

C. Vulnerability Response

D. Design and Code Reviews

E. Applying Processes and Practices

F. Risk Analysis

VI. Security Testing

A. Testing as Lifecycle Process

B. Testing Planning and Documentation

C. Testing Tools And Processes
1. Principles
2. Reviews
3. Testing
4. Tools

D. Static and Dynamic Code Analysis

E. Testing Practices
1. Authentication Testing
2. Session Management Testing
3. Data Validation Testing
4. Denial Of Service Testing
5. Web Services Testing
6. Ajax Testing

VII. Appendix: Security Design Patterns

A. Design Patterns Introduction

B. Web Application Security Design Patterns
1. Authentication Enforcer
2. Authorization Enforcer
3. Intercepting Validator
4. Secure Base Action
5. Secure Logger
6. Secure Pipe
7. Secure Service Proxy
8. Intercepting Web Agent