Course Outline

ProTech Professional Technical Services, Inc.

protechtraining.com

OWASP and Security Concepts for Software Developers

Course Summary

Description

This course for Developers is designed to expose software developers to the key security concepts that they need to know to gain a full appreciation of secure coding. This is mostly a language-agnostic course that focuses on the concepts, techniques, and mechanisms required to secure data and to create secure software that enforces and maintains data protection. Most developers are aware of some of these concepts, but they do not fully appreciate the significance of each in relation to the others, and how these topics ultimately affect their ability to evaluate and implement secure coding practices. Any factors that affect software security should be carefully considered and fully understood. This course helps ensure that developers are adequately equipped to make properly informed choices during each coding project.

Objectives

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

- Identify software security design principles and how to apply them.
- Identify the importance of Defense-in-Depth.
- Create manageable security policies that can actually be implemented.
- Protect electronic data and software systems.
- Apply common design patterns and best practices.
- Utilize Certificates, Authentication, Authorization, and Encryption.
- Demonstrate an understanding of Cryptography and how it can be used to protect data.
- Implement defenses to avoid common threats.
- Recognize the necessity of a secure infrastructure and culture.
- Understand and defend against the OWASP top 10.

Topics

- Security Goals
- Secure Systems Design
- Secure Design Principles
- Worms and Other Malware
- Buffer Overflows
- Client-State Manipulation
- SQL Injection
- Password Security
- Jumping into the OWASP Top 10
- A1: Injection
- A2: Broken Authentication
- A3: Sensitive Data Exposure
- A4: XML External Entities (XXE)
- A5: Broken Access Control
- A6: Security Misconfiguration
- A7: Cross Site Scripting (XSS)
- Cross-Domain Security in Web Applications
- Symmetric Key Cryptography
- Key Management and Exchange
- MACs and Signatures
- Appendix A: Hacking and Penetration Testing
- API Security – If time allows

Audience

This course is designed for Developers with Java programming experience.

Prerequisites

Before taking this course, students should have some development experience to complete the lab exercises.

Duration

Five days

protechtraining.com

OWASP and Security Concepts for Software Developers

Course Outline

OWASP and Security Concepts for Software Developers

Course Outline (cont'd)

XI. A2: Broken Authentication

A. Weak Authentication Data
B. Protecting Authentication Data
C. Protecting Authentication Services
D. Effective Credential Management
E. Effective Multi-Factor Authentication
F. Handling Passwords on Server Side
G. Potential Demonstration/Lab: Defending Authentication

XII. A3: Sensitive Data Exposure

A. Protecting Data Can Mitigate Impact of Exploit
B. Regulatory Considerations
C. Establishing an Asset Inventory
D. At Rest Data Handling
E. In Motion Data Handling
F. In Use Data Handling
G. Potential Demonstration/Lab: Defending Sensitive Data

XIII. A4: XML External Entities (XXE)

A. Recognizing XML Processing: Direct, REST, SOAP, etc.
B. Challenges of Safe XML Parsing
C. Managing External Entity Resolution
D. XSLT Processing Challenges
E. Safe XML Processing
F. Potential Demonstration/Lab: Safe XML Processing

XIV. A5: Broken Access Control

A. Access Control and Trust Boundaries
B. Excessive Privileges
C. Insufficient Flow Control
D. Unprotected API Resource Access
E. JWTs, Sessions and Session Management
F. Single Sign-on (SSO)
G. Potential Demonstration/Lab: Enforcing Access Control

XV. A6: Security Misconfiguration

A. System Hardening: IA Mitigation
B. Application Whitelisting
C. Principle of Least Privileges in Real Terms
D. Secure Configuration Baseline
E. Error-Handling Issues

XVI. A7: Cross Site Scripting (XSS)

A. XSS Patterns
B. Stored XSS
C. Reflected XSS
D. DOM XSS
E. Best Practices for Untrusted Data
F. Potential Demonstration/Lab: Defending Against XSS

XVII. Cross-Domain Security in Web Applications

A. Interaction Between Web Pages from Different Domains
B. Attack Patterns
C. Cross-Site Request Forgery (CSRF)
D. Cross-Site Script Inclusion (XSSI)
E. Cross-Site Scripting (XSS)
F. Preventing CSRF
G. Preventing XSSI
H. Preventing XSS

XVIII. Symmetric Key Cryptography

A. Introduction to Encryption
B. Stream Cipher

XIX. Asymmetric Key Cryptography

A. Why Asymmetric Key Cryptography?
B. RSA
C. Elliptic Curve Cryptography (ECC)
D. Symmetric vs Asymmetric Key Cryptography
E. Certificate Authorities
F. Identity-Based Encryption (IBE)
G. Authentication with Encryption

OWASP and Security Concepts for Software Developers

Course Outline (cont'd)

XX. Key Management and Exchange

A. Types of Keys
B. Identity Keys
C. Conversation or Session Keys
D. Integrity Keys
E. Key Generation
F. Random Number Generation
G. The rand() function
H. Random Device Files
I. Random APIs
J. Key (Secret) Storage
K. Keys in Source Code
L. Storing the Key in a File on Disk
M. "Hard to Reach" Places
N. Storing Secrets in External Devices
O. Key Agreement and Exchange

XXI. MACs and Signatures

A. Secure Hash Functions
B. Message Authentication Codes (MACs)
C. CBC MACs
D. HMAC
E. Signatures
F. Certificates and Certificate Authorities (CAs)
G. Signing and Verifying
H. Registration Authorities (RAs)
I. Web of Trust
J. Attacks Against Hash Functions
K. SSL

XXII. Appendix A: Hacking and Penetration Testing

XXIII. API Security – If time allows