SECURING .NET WEB APPLICATIONS (TT8320-N) Training

Duration: **4 Day**

IT Academy

info@bilginc.com
bilginc.com

Overview

Covering OWASP Top Ten, Web Services, Rich Interfaces and more

In this course, you will thoroughly examine best practices for defensively coding .NET web applications, including XML processing and web services. You will repeatedly attack and then defend various assets associated with a fully-functional web application. This hands-on approach drives home the mechanics of how to secure .NET web applications in the most practical of terms. This workshop is a companion course with several developer-oriented courses and seminars. Although this edition of the course is .NET-specific, it may also be presented using JEE or other programming languages.

PCI Compliant Developer Training: Version 3.0 of the Payment Card Information Data Security Standard (PCI-DSS) and the Payment Application Data Security Standard (PA-DSS) have placed an increased emphasis on information security training and awareness. This class can help meet the annual training requirements for your developers and vendors. This secure coding training addresses common coding vulnerabilities in software development processes. This training is used by one of the principle participants in the PCI DSS. Having passed multiple PCI audits, this course has been shown to meet the PCI requirements. The specification of those training requirements are detailed in 6.5.1 through 6.5.10 on pages 55 through 59 of the PCI DSS Requirements 3.0 document dated November 2013.

Who Should Attend?

This intermediate-level .NET programming course is designed for developers who wish to get up and running on developing well-defended software applications.

SECURING .NET WEB APPLICATIONS (TT8320-N) Training

Duration: **4 Day**

IT Academy

info@bilginc.com
bilginc.com

What You Will Learn

- Potential sources for untrusted data
- Consequences for not properly handling untrusted data such as denial of service, cross-site scripting, and injections
- 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
- Vulnerabilities of associated with authentication and authorization
- Be able to detect, attack, and implement defenses for authentication and authorization functionality and services
- Dangers and mechanisms behind Cross-Site Scripting (XSS) and Injection attacks
- Detect, attack, and implement defenses for authentication and authorization functionality and services
- Concepts and terminology behind defensive, secure, coding
- Threat Modeling as a tool in identifying software vulnerabilities based on realistic threats against assets
- Static code reviews and dynamic application testing for uncovering vulnerabilities in web applications
- Design and develop strong, robust authentication and authorization implementations within the context of .NET
- Fundamentals of XML Digital Signature and XML Encryption as well as how they are used within the web services arena
- Detect, attack, and implement defenses for XML-based services and functionality
- Techniques and measures that can used to harden web and application servers as well as other components in your infrastructure

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

IT Academy

info@bilginc.com

bilginc.com

    - 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

3. Vulnerabilities

- Unvalidated Input
    - Buffer Overflows
    - Integer Arithmetic Vulnerabilities
    - Unvalidated Input: From the Web
    - Defending Trust Boundaries
    - Whitelisting vs Blacklisting
- Overview of Regular Expressions
    - Regular Expressions
    - Working With Regexes in .NET
    - Applying Regular Expressions
- Broken Access Control
    - Access Control Issues
    - Excessive Privileges
    - Insufficient Flow Control
    - Unprotected URL/Resource Access
    - Examples of Shabby Access Control
    - Session and Session Management
- Broken Authentication
    - Broken Quality/DoS
    - Authentication Data
    - Username/Password Protection
    - Exploits Magnify Importance
    - Handling Passwords on Server Side
    - Single Sign-on (SSO)
- Cross Site Scripting (XSS)
    - Persistent XSS
    - Reflective XSS
    - Best Practices for Untrusted Data
- Injection
    - Injection Flaws
    - SQL Injection Attacks Evolve
    - Drill Down on Stored Procedures
    - Other Forms of Injection
    - Minimizing Injection Flaws
- Error Handling and Information Leakage
    - Fingerprinting a Web Site
    - Error-Handling Issues
    - Logging In Support of Forensics
    - Solving DLP Challenges
- Insecure Data Handling
    - Protecting Data Can Mitigate Impact

IT Academy

info@bilginc.com

bilginc.com

    - In-Memory Data Handling
    - Secure Pipes
    - Failures in the SSL Framework Are Appearing
- Insecure Configuration Management
    - System Hardening: IA Mitigation
    - Application Whitelisting
    - Least Privileges
    - Anti-Exploitation
    - Secure Baseline
- Direct Object Access
    - Dynamic Loading
    - Race Conditions
    - Direct Object References
- Spoofing, CSRF, and Redirects
    - Name Resolution Vulnerabilities
    - Fake Certs and Mobile Apps
    - Targeted Spoofing Attacks
    - Cross Site Request Forgeries (CSRF)
    - CSRF Defenses are Entirely Server-Side
    - Safe Redirects and Forwards

4. Best Practices

- .NET Issues and Best Practices
    - Manage Code and Buffer Overflows
    - .NET Permissions
    - ActiveX Controls
    - Proper Exception Handling
- 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

5. Defending XML, Services, and Rich Interfaces

- 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
    - Web Service Attacks
    - Web Service Appliance/Gateways
- Defending Rich Interfaces and REST

IT Academy
info@bilginc.com
bilginc.com

- How Attackers See Rich Interfaces
- Attack Surface Changes When
- Moving to Rich Interfaces
- Bridging and its Potential Problems
- Three Basic Tenets for Safe Rich Interfaces
- OWASP REST Security Recommendations