

## Securing Web Applications, Services and Servers: Hands-On - 4 Days

### *Course 940 Overview*

- You Will Learn How To**
- Implement and test secure web applications in your organization
  - Identify, diagnose and correct the most serious web application vulnerabilities
  - Configure a web server to encrypt web traffic with HTTPS
  - Protect Ajax-powered Web 2.0 applications
  - Secure XML web services with WS-Security
  - Audit web application security with manual and automated scanning

**Course Benefits** Cybersecurity is a serious challenge today as attackers specifically target web application vulnerabilities. These vulnerabilities can be exploited to obtain confidential information and compromise organizational integrity. As a result, organizations must integrate robust security measures into the web application development process. This course provides in-depth, hands-on experience securing web-based applications and host servers.

**Who Should Attend** Those who want to implement, test and deploy secure web applications. Basic knowledge of web application operation and web server administration is assumed. Web application development and security knowledge are helpful but not required.

**Hands-On Training** Throughout this course, extensive hands-on exercises based on an evolving case study provide you with practical experience in securing applications. Exercises include:

- Creating a trust boundary with proper input validation
- Avoiding cross-site scripting (XSS) and cross-site request forgery (CSRF/XSRF)
- Preventing SQL injection vulnerabilities
- Implementing URL access restrictions
- Detecting unauthorized file system modification
- Enabling HTTPS on a web server
- Protecting web services with WS-Security
- Identifying vulnerabilities with an application scanner

# Securing Web Applications, Services and Servers: Hands-On - 4 Days

## Course 940 Outline

### Setting the Stage

- Defining threats to your web assets
- Surveying the legal landscape and privacy issues
- Exploring common vulnerabilities

### Establishing Security Fundamentals

#### Modeling web security

- Achieving Confidentiality, Integrity and Availability (CIA)
- Performing authentication and authorization

### Encrypting and hashing

- Distinguishing public- and private-key cryptography
- Verifying message integrity with message digests, digital signatures and digital certificates

### Augmenting Web Server Security

#### Configuring security for HTTP services

- Managing software updates
- Restricting HTTP methods

### Securing communication with SSL/TLS

- Obtaining and installing server certificates
- Enabling HTTPS on the web server
- Protecting the exchange of credentials

### Detecting unauthorized modification of content

- Configuring permissions correctly
- Scanning for file-system changes

### Implementing Web Application Security

#### Employing OWASP resources

- The Open Web Application Security Project (OWASP) Top Ten
- Recognizing cybersecurity risks
- Remediating identified vulnerabilities

### Securing database and application interaction

- Uncovering and preventing SQL injection
- Defending against an insecure direct object reference
- Limitations of encrypting database content

### Managing session authentication

- Protecting against session ID hijacking
- Enforcing URL access control
- Blocking cross-site request forgery

### Controlling information leakage

- Displaying sanitized error messages to the user
- Handling request and page faults

### Performing input validation

- Establishing trust boundaries
- Revealing and removing the threat of cross-site scripting (XSS)
- Exposing the dangers of client-side validation
- Preventing E-shoplifting

### Enhancing Ajax Security

#### Ajax features

- Identifying core Ajax components
- Exchanging information asynchronously

### Assessing risks and evaluating threats

- Managing unpredictable interactions
- Exposing JSON vulnerabilities

### Securing XML Web Services

#### Diagnosing XML vulnerabilities

- Identifying nonterminated tags and field overflows
- Uncovering web service weaknesses

### Protecting the SOAP message exchange

- Validating input with an XML schema
- Encrypting exchanges with HTTPS
- Implementing WS-Security with a framework
- Authenticating access to web services

### Scanning Applications for Weaknesses

#### Operating and configuring scanners

- Matching patterns to identify faults
- "Fuzzing" to discover new or unknown vulnerabilities

### Detecting application flaws

- Scanning applications remotely
- Strategies for testing and scanning
- Testing web applications with Netcat, Cryptcat and Wget
- Intercepting traffic with OWASP WebScarab

### Best Practices for Web Security

#### Adopting standards

- Reducing risk by implementing proven architectures

- Handling personal and financial data
- Developing guidelines for logging

### Managing network security

- Modeling threats to reduce risk
- Integrating applications with your network architecture