SHI | TRAINING & ADOPTION

# Course Outline

## Securing Web Applications | Explore OWASP Top Ten, Bug Hunting, Bug Stomping & More
## Course TT8120: 2 days Instructor Led

### About this course

Security experts agree that the least effective approach to security is "penetrate and patch". It is far more effective to "bake" security into an application throughout its lifecycle. After spending significant time examining a poorly designed (from a security perspective) web application, developers are ready to learn how to build secure web applications starting at project inception. The final portion of this course builds on the previously learned mechanics for building defenses by exploring how design and analysis can be used to build stronger applications from the beginning of the software lifecycle.

The Secure Web Application Development Overview is geared for web developers and technical stakeholders who need to produce secure web applications, integrating security measures into the development process from requirements to deployment and maintenance. This overview-level course explores core concepts and challenges in web application security, showcasing current, real world examples that illustrate the potential consequences of not following these best practices.

This course is also PCI Compliant.

### Audience profile

This is an overview-level, lecture and demonstration style course, designed to provide technical application project stakeholders with a first-look or baseline understanding of how to develop well defended web applications. Real-world programming experience is highly recommended for code reviews, but not required.

Take After: We offer a variety of introductory through advanced security, development, project management, engineering, architecture and design courses that serve as an excellent follow on to this course. Please inquire for details.

- Java, Node.js, C++ or .Net oriented Web Application Security hands-on training
- Refresher training for updated skills or to fulfill PCI compliant requirements

### At course completion

After completing this course, students will be able to:

- Understand the concepts and terminology behind defensive, secure coding including the phases and goals of a typical exploit
- Ensure that any hacking and bug hunting is performed in a safe and appropriate manner
- Identify defect/bug reporting mechanisms within their organizations
- Avoid common mistakes that are made in bug hunting and vulnerability testing
- Develop an appreciation for the need and value of a multilayered defense in depth
- 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
- Understand the vulnerabilities of associated with authentication and authorization
- 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
- Detect, attack, and implement defenses against XSS and Injection attacks
- Understand the risks associated with XML processing, file uploads, and server-side interpreters and how to best eliminate or mitigate those risks
- Learn the strengths, limitations, and use for tools such as code scanners, dynamic scanners, and web application firewalls (WAFs)
- Understand techniques and measures that can used to harden web and application servers as well as other components in your infrastructure
- Identify resources to use for ongoing threat intelligence
- Plan next steps after completion of this training

SHI | TRAINING & ADOPTION

# Course Outline

## Course Outline

**Session: Bug Hunting Foundation**

**Lesson: Why Hunt Bugs?**
- The Language of Cybersecurity
- The Changing Cybersecurity Landscape
- AppSec Dissection of SolarWinds
- The Human Perimeter
- Interpreting the 2021 Verizon Data Breach Investigation Report
- Lab: Case Study in Failure

**Lesson: Safe and Appropriate Bug Hunting/Hacking**
- Working Ethically
- Respecting Privacy
- Bug/Defect Notification
- Bug Bounty Programs
- Bug Hunting Mistakes to Avoid

**Session: Moving Forward From Hunting Bugs**

**Lesson: Removing Bugs**
- Open Web Application Security Project (OWASP)
- OWASP Top Ten Overview
- Web Application Security Consortium (WASC)
- CERT Secure Coding Standards
- Microsoft Security Response Center
- Software-Specific Threat Intelligence

**Session: Foundation for Securing Web Applications**

**Lesson: 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
- AppSec Dissection of the Verkada Exploit

**Session: Bug Stomping 101**

**Lesson: Unvalidated Data**
- Buffer Overflows
- Integer Arithmetic Vulnerabilities
- Defining and Defending Trust Boundaries
- Rigorous., Positive Specifications
- Whitelisting vs Blacklisting
- Challenges: Free-Form Text, Email Addresses, and Uploaded Files
- Demo: Defending Trust Boundaries

# Course Outline

**Lesson: A1: Injection**
- Injection Flaws
- SQL Injection Attacks Evolve
- Drill Down on Stored Procedures
- Other Forms of Injection
- Minimizing Injection Flaws
- Demo: Defending Against SQL Injection

**Lesson: A2: Broken Authentication**
- Quality and Protection of Authentication Data
- Handling Passwords on Server Side
- Sessions and Session Management
- HttpOnly and Security Headers
- Demo: Defending Authentication

**Lesson: A3: Sensitive Data Exposure**
- Hidden Data Stores
- Evolving Privacy Considerations
- Options for Protecting Data
- In-Memory Data Handling
- Transport and Message Level Security – What Each Addresses

**Lesson: A4: XML External Entities (XXE)**
- XML-Related Challenges
- XML Parser Coercion
- Resolution of External References
- Safe XML Processing
- Demo: Safe XML Processing

**Lesson: A5: Broken Access Control**
- Access Control Issues
- Excessive Privileges
- Insufficient Flow Control
- Unprotected URL/Resource Access
- Demo: Unsafe Direct Object References
- Lab: Spotlight: Verizon

Session: Bug Stomping 102

**Lesson: A6: Security Misconfiguration**
- System Hardening: IA Mitigation
- Application Whitelisting
- Least Privileges
- Anti-Exploitation
- Secure Baseline

**Lesson: A7: Cross Site Scripting (XSS)**
- XSS Patterns
- Persistent XSS
- Reflective XSS
- DOM-Based XSS

# Course Outline

- Best Practices for Untrusted Data
- Demo: Defending Against XSS

**Lesson: A8/9: Deserialization/Vulnerable Components**
- Deserialization Issues
- Identifying Serialization and Deserializations
- Vulnerable Components
- Software Inventory
- Managing Updates
- AppSec Dissection of Ongoing Microsoft Exchange Exploits
- Lab: Spotlight: Equifax

**Lesson: A10: Insufficient Logging and Monitoring**
- Fingerprinting a Web Site
- Error-Handling Issues and Fixes
- Best Practices for Determining What to Log
- Forensics and Data Loss Prevention

**Lesson: Spoofing and CSRF**
- Name Resolution Vulnerabilities
- Fake Certs and Mobile Apps
- Targeted Spoofing Attacks
- Cross Site Request Forgeries (CSRF)
- CSRF Defenses
- Demo: Cross-Site Request Forgeries

**Session: Moving Forward**

**Lesson: Applications: What Next?**
- Common Vulnerabilities and Exposures
- CWE/SANS Top 25 Most Dangerous SW Errors
- Strength Training: Project Teams/Developers
- Strength Training: IT Organizations
- Lab: Recent Incidents
- Lab: Spotlight: Capital One

**Lesson: Making Application Security Real**
- Cost of Continually Reinventing
- Leveraging Common AppSec Practices and Control
- Paralysis by Analysis
- Actional Application Security
- Additional Tools for the Toolbox

**Bonus Topics / Time Permitting**

**Lesson: SDL Overview**
- Attack Phases: Offensive Actions and Defensive Controls
- Secure Software Development Processes
- Shifting Left
- Actionable Items Moving Forward

SHI | TRAINING & ADOPTION

# Course Outline

**Lesson: SDL in Action**

- Risk Escalators
- Risk Escalator Mitigation
- SDL Phases
- Actions for each SDL Phase
- SDL Best Practices