PHP Secure Programming

3-Day Course

Course Syllabus

AppSec Labs Ltd.
info@appsec-labs.com | https://appsec-labs.com | 23 HaTa'as St., Kfar Saba 44641 Israel | T: +972-9-7485005 | F: +972-9-7730595

PHP Secure Coding
3-day course

Course description

Secure web programming is the best defense against hackers. This multilayered hands-on course will demonstrate live real time hacking methods, analyze the code deficiency that enabled the attack and most importantly, teach how to prevent such vulnerabilities by adopting secure coding best practices in order to bullet-proof your application.

The methodology of the cycle of knowledge is as follows: understand, identify, prevent. This methodology presents the student with analytical tools to keep a deeper understanding of coding vulnerabilities and implement security countermeasures in different areas of the software development lifecycle. The hands on labs will enable the student to get a firsthand experience of the hackers' world and what could be done to stop him. Using sound programming techniques and best practices shown in this course, you will be able to produce high-quality code that stands up to attack.

The course covers major security principles in PHP programming vulnerabilities, and known security issues in web applications

Target audience

Members of the software development team:

- PHP Developers
- Designers & architects

Prerequisites

Before attending this course, students should be familiar with:

- Background knowledge of PHP
- Apache
- MySQL & SQL language
- **Students should bring their own laptop**

Course topics

Day 1

Authentication

- What is authentication
- Store password securely
- Hashing
- Brute force
- Dictionary attack
- Anti-Automation
- Account lockout
- User numeration
- Basic & Digest authentication
- Windows integration
- Form based authentication
- **Hands-on Lab**

Authorization

- Client side authorization
- Forceful browsing
- UI based security
- Parameter tampering
- Insecure direct object reference
- File authorization
- URL authorization
- ACL (Access Control List)
- RBAC (Role based ACL)
- **Hands-on Lab**

Input Validation

- Client side authorization
- Injection Flaws
- OS Command Injection
- SQL Injection
- Parameterized queries
- Stored procedures
- XPATH Injection
- LDAP Injection
- Strong typing
- Blacklist vs. Whitelist validation
- Regular expressions (Regex)
- Using ESAPI PHP
- **Hands-on Lab**

Day 2

Output Encoding

- Reflected / Stored Cross-site scripting
- XSS threats
- Encoding types
- HTML Special Characters
- ESAPI library
- XSS prevention cheat sheet
- **Hands-on Lab**

Browser Manipulation

- Cross-Site Request Forgery (CSRF)
- Anti CSRF token
- Open redirect
- Clickjacking
- Browser's cache
- Session management
- Cookie's properties
- Session fixation
- **Hands-on Lab**

File Handling

- Session fixation
- Directory traversal
- Canonicalization
- File extension handling
- LFI by PHP filter
- Null-byte injection
- Uploaded files backdoors
- Filenames threats
- Directory listing
- **Hands-on Lab**

Day 3

Data confidentiality

- Insecure communication
- Secure traffic enforcement
- Insecure storage
- Symmetric encryption
- A-Symmetric encryption
- Hash functions
- Digital signatures
- **Hands-on Lab**

Exception management & Logging

- Information disclosure
- Error settings in Apache / Tomcat
- Custom error pages
- Page vs. Application level error handling
- Error handling strategy
- To log or not to log
- **Hands-on Lab**

Business Logic

- Logical attacks
- Flow bypassing
- Replay attacks
- Abuse of functionality
- **Hands-on Lab**

Misconfiguration

- Security settings in php.ini
- Protection connection strings
- Avoiding hard coded secrets

Secure coding with HTML5 & JS

- Introduction to HTML 5
- Protecting stored data at the client side
- Client side Web SQL Database Security
- Offline cache poisoning
- SQL injection in HTML5
- XSS in HTML5
- Securing cross origin requests
- Sandboxed iframe
- Cross-Document Messaging
- attribute handling
- Client side validation in HTML5
- **Hands-on Lab**