

Certified Secure Web Application Engineer

COURSE OVERVIEW

Course Name: Certified Secure Web Application Engineer

Duration: 4 days

Format:

Instructor-led classroom

Prerequisites:

- A minimum of 24 months' experience in software technologies & security
- Sound knowledge of networking
- At least one coding Language
- Linux understanding
- Open shell

Student Materials:

- Student Workbook
- Student Lab Guide
- Exam prep guide

Certification Exam:

Mile2's CSWAE- Certified Secure Web Application Engineer

CPEs: 32 Hours

WHO SHOULD ATTEND?

- Coders
- Web Application Engineers
- IS Managers
- Application Engineers
- Developers
- Programmers

Organizations and governments fall victim to internet based attacks every day. In many cases, web attacks could be thwarted but hackers, organized criminal gangs, and foreign agents are able to exploit weaknesses in web applications. The Secure Web programmer knows how to identify, mitigate and defend against all attacks through designing and building systems that are resistant to failure. The secure web application developer knows how to develop web applications that aren't subject to common vulnerabilities, and how to test and validate that their applications are secure, reliable and resistant to attack. The vendor neutral **Certified Secure Web Application Engineer** certification provides the developer with a thorough and broad understanding of secure application concepts, principles and standards. The student will be able to design, develop and test web applications that will provide reliable web services that meet functional business requirements and satisfy compliance and assurance needs.

The **Certified Secure Web Application Engineer** course is delivered by high level OWASP experts and students can expect to obtain real world security knowledge that enables them to recognize vulnerabilities, exploit system weaknesses and help safeguard against application threats.

Application & Secure Code Career



C)SSTM
CERTIFIED SECURITY
SENTINEL



C)ISSOTM
Certified Information Systems
Security Officer



C)SWAETM
CERTIFIED SECURE WEB
APPLICATION ENGINEER *

All combos include:

- Electronic Book (Workbook/Lab guide)
- Exam Prep Questions
- Exam

ACCREDITATIONS



NICCSTM

NATIONAL INITIATIVE FOR
CYBERSECURITY CAREERS AND STUDIES



is **ACCREDITED** by the **NSA CNSS 4011-4016**
is **MAPPED** to NIST/Homeland Security NICCS's Cyber Security Workforce Framework
is **APPROVED** on the **FBI Cyber Security Certification Requirement list (Tier 1-3)**

UPON COMPLETION

Upon completion, **Certified Secure Web Application Engineer** students will be able to establish industry acceptable auditing standards with current best practices and policies. Students will also be prepared to competently take the C)SWAE exam.

EXAM INFORMATION

The **Certified Secure Web Application Engineer** exam will take 2 hours and consist of 100 multiple choice questions.

COURSE CONTENT

- Module 1: Web Application Security
- Module 2: OWASP TOP 10
- Module 3: Threat Modeling & Risk Management
- Module 4: Application Mapping
- Module 5: Authentication and Authorization attacks
- Module 6: Session Management attacks
- Module 7: Application Logic attacks
- Module 8: Data Validation
- Module 9: AJAX attacks
- Module 10: Code Review and Security Testing
- Module 11: Web Application Penetration Testing
- Module 12: Secure SDLC
- Module 13: Cryptography

LAB CONTENT



- Module 1 – Environment Setup and Architecture
- Module 2 – OWASP TOP 10 2013
- Module 3 – Threat Modeling



Module 04 – Application Mapping & Analysis

Module 5 – Authentication and Authorization attacks

Module 06 - Session Management attacks

Module 9 – AJAX Security

Module 10 – Code Review and Security Testing

Lab 10-1 – Code Review

Lab 10-2 Security Test Scripts

Lab 10-3 Writing Java Secure Code

Annex 11: Alternatives Labs

Lab 11-1: WebGoat & WebScarab

Lab 11-2: WebGoat - Cross Site Request Forgery (CSRF)

Lab 11-3: Missing Function Level Access Control

Lab 11-4: Perform Forced Browsing Attacks

DETAILED OUTLINE

- **Module 1: Web Application Security**

Web Application Security
Web Application Technologies and Architecture
Secure Design Architecture
Application Flaws and Defense Mechanisms
Defense In-Depth
Secure Coding Principles

- **Module 2: OWASP TOP 10**

The Open Web Application Security Project (OWASP)
OWASP TOP 10 2013

- **Module 3: Threat Modeling & Risk Management**

Threat Modeling Tools & Resources
Identify Threats
Identify Countermeasures
Choosing a Methodology
Post Threat Modeling
Analyzing and Managing Risk
Incremental Threat Modeling
Identify Security Requirements
Understand the System
Root Cause Analysis

- **Module 4: Application Mapping**

Application Mapping
Web Spiders
Web Vulnerability Assessment
Discovering other content
Application Analysis
Application Security Toolbox
Setting up a Testing Environment

- **Module 5: Authentication and Authorization attacks**

Authentication
Different Types of Authentication (HTTP, Form)
Client Side Attacks
Authentication Attacks
Authorization
Modeling Authorization
Least Privilege
Access Control
Authorization Attacks
Access Control Attacks
User Management
Password Storage
User Names
Account Lockout
Passwords
Password Reset
Client-Side Security
Anti-Tampering Measures
Code Obfuscation
Anti-Debugging

- **Module 6: Session Management attacks**

Session Management
Attacks
Session Hijacking
Session Fixation
Environment Configuration
Attacks

- **Module 7: Application Logic attacks**

Application Logic Attacks
Information Disclosure
Exploits
Data Transmission Attacks

Module 8: Data Validation

Input and Output Validation
Trust Boundaries
Common Data Validation Attacks
Data Validation Design
Validating Non-Textual Data
Validation Strategies & Tactics
Errors & Exception Handling

- Structured Exception Handling
- Designing for Failure
- Designing Error Messages
- Failing Securely

Module 9: AJAX attacks

AJAX Attacks
Web Services Attacks
Application Server Attacks

Module 10: Code Review and Security Testing

Insecure Code Discovery and Mitigation
Testing Methodology
Client Side Testing
Session Management Testing
Developing Security Testing Scripts
Pen testing a Web Application

Module 11: Web Application Penetration Testing

Insecure Code Discovery and Mitigation
Benefits of a Penetration Test
Current Problems in WAPT
Learning Attack Methods
Methods of Obtaining Information
Passive vs. Active Reconnaissance

Footprinting Defined
Introduction to Port Scanning
OS Fingerprinting
Web Application Penetration Methodologies
The Anatomy of a Web Application Attack
Fuzzers

Module 12: Secure SDLC

Secure-Software Development Lifecycle (SDLC) Methodology
Web Hacking Methodology

Module 13: Cryptography

Overview of Cryptography
Key Management
Cryptography Application
True Random Generators (TRNG)
Symmetric/Asymmetric Cryptography
Digital Signatures and Certificates
Hashing Algorithms
XML Encryption and Digital Signatures
Authorization Attacks

NOTE: Student will use Kali Linux

DETAILED LAB DESCRIPTION



Module 1 – Environment Setup and Architecture

- Exercise 1 – VM Image Preparation
- Exercise 2 – Checking Network connectivity between all VMs
- Exercise 3 – Discovering your class share (Optional, ask the Instructor)
- Exercise 4 – Navigating Linux Attack v3
- Exercise 5 – Proxy Setup - Setting up Burp Suite
- Exercise 6 – Setting up Paros
- Exercise 7 – Setting up WebScrab

Module 2 – OWASP TOP 10 2013

- Exercise 1- Injection Flaws - SQL Injection (AltoroMutual banking site)
- Exercise 2- Injection Flaws – String SQL Injection (OWASP Broken Apps WebGoat)
- Exercise 3- Cross Site Scripting (XSS)
- Exercise 4 - Cross Site Request Forgery (CSRF)

Module 3 – Threat Modeling

- Exercise 1 – Application Risk Assessment
- Exercise 2: Define the Entry Points
- Exercise 3: Define the Assets
- Exercise 4: Define User Access
- Exercise 5: Identify and Rate Risks
- Exercise 6: Identify Security Controls
- Exercise 7: Identify Threats

Module 04 – Application Mapping & Analysis

- Exercise 1 - Enumerating Content and Functionality
- Exercise 2 - User-Directed Spidering
- Exercise 3 - Discovering hidden content
- Exercise 4 - Brute-Force Techniques brute force DVWA
 - a. Form Based Authentication
 - b. Attacking Web Authentication

Module 5 – Authentication and Authorization attacks

- Exercise 1 - Missing Function Level Access Control
- Exercise 2 - Sensitive Data Exposure
- Exercise 3 - Security Misconfiguration
- Exercise 4 - Using Components with Known Vulnerabilities

Module 06 - Session Management attacks

- Exercise 1 - Hijack a Session
- Exercise 2 - Spoof an Authentication Cookie
- Exercise 3 - Session Fixation

Exercise 4 - Broken Authentication and Session Management (AltoroMutual banking)

Module 9 – AJAX Security

Exercise 1: Same Origin Policy Protection

Exercise 2: DOM-Based cross-site scripting

Exercise 3: Client Side Filtering

Module 10 – Code Review and Security Testing

Lab 10-1 – Code Review

Exercise 1: Account Retriever

Exercise 2: FileUpload

Exercise 3: XMLHelper

Lab 10-2 Security Test Scripts

Exercise 1: Create Test Scripts

Lab 10-3 Writing Java Secure Code

Annex: Alternatives Labs

Lab 11-1: WebGoat & WebScarab

Exercise 11-1.1: Logging into WebGoat

Exercise 11-1.2: Running WebScarab

Exercise 11-1.3: Manipulating Data

Lab 11-2: WebGoat - Cross Site Request Forgery (CSRF)

Lab 11-3: Missing Function Level Access Control

Lab 11-4: Perform Forced Browsing Attacks