Java Certified Application Security Engineer (JAVA CASE)

Description

Become an Expert in Java Application Security

Application security is a priority in modern development. With the Java Certified Application Security Engineer (JAVA CASE) certification, you will learn to design, develop, and maintain secure applications, while addressing the growing cybersecurity demands. This training covers all phases of the software development life cycle (SDLC), with a focus on securing applications from the design phase.

Whether you are a developer, analyst, or software architect, this course will equip you with the necessary skills to secure applications in complex environments. You will learn to identify common security flaws, apply secure coding best practices, and use security testing tools. Don’t miss this opportunity to become a leader in application security.

Course Content
Module 1: Understanding Application Security, Threats and Attacks

- What is a Secure Application
- Need for Application Security
- Most Common Application Level Attacks
- Why Applications become Vulnerable to Attacks
- What Constitutes Comprehensive Application Security
- Insecure Application: A Software Development Problem
- Software Security Standards, Models and Frameworks

Module 2: Security Requirements Gathering

- Importance of Gathering Security Requirements
- Security Requirement Engineering (SRE)
- Abuse Case and Security Use Case Modeling
- Abuser and Security Stories
- Security Quality Requirements Engineering (SQUARE)
- Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE)

Module 3: Secure Application Design and Architecture

- Relative Cost of Fixing Vulnerabilities at Different Phases of SDLC
- Secure Application Design and Architecture
- Goal of Secure Design Process
- Secure Design Actions
- Secure Design Principles
- Threat Modeling
- Decompose Application
- Secure Application Architecture

Module 4: Secure Coding Practices for Input Validation

- Input Validation Pattern
- Validation and Security Issues
- Impact of Invalid Data Input
- Data Validation Techniques
- Input Validation using Frameworks and APIs
- Open Source Validation Framework for Java
- Servlet Filters Validation Filters for Servlet
- Data Validation using OWASP ESAPI
- Data Validation: Struts Framework
- Data Validation: Spring Framework
- Input Validation Errors

Module 5: Secure Coding Practices for Authentication and Authorization

- Introduction to Authentication
- Types of Authentication
- Authentication Weaknesses and Prevention
- Introduction to Authorization
- Access Control Model
- EJB Authorization
- Java Authentication and Authorization (JAAS)
- Java EE Security
- Authorization Common Mistakes and Countermeasures
- Authentication and Authorization in Spring Security Framework
- Defensive Coding Practices against Broken Authentication and Authorization

Module 6: Secure Coding Practices for Cryptography

- Java Cryptographic
- Encryption and Secret Keys
- Cipher Class
- Digital Signatures
- Secure Socket Layer (SSL)
- Key Management
- Digital Signatures
- Signed Code Sources
- Hashing
- Java Card Cryptography
- Spring Security: Crypto Module
- Do's and Don'ts in Java Cryptography

- Best Practices for Java Cryptography

Module 7: Secure Coding Practices for Session Management

- Session Management
- Session Tracking
- Session Management in Spring Security
- Session Vulnerabilities and their Mitigation Techniques
- Best Practices and Guidelines for Secured Sessions Management
- Checklist to Secure Credentials and Session IDs
- Guidelines for Secured Session Management

Module 8: Secure Coding Practices for Error Handling

- Introduction to exceptions
- Erroneous Exceptional Behaviors
- Do's and Don'ts in Error Handling
- Spring MVC Error Handling
- Exception Handling in Struts 2
- Best Practices for Error Handling
- Introduction to Logging
- Logging using Log4j
- Secure Coding in Logging

Module 9: Static and Dynamic Application Security Testing (SAST and DAST)

- Static Application Security Testing
- Manual Secure Code Review for Most Common Vulnerabilities
- Code Review: Checklist Approach
- SAST Finding
- SAST Report
- Dynamic Application Security Testing
- Automated Application Vulnerability Scanning Tools
- Proxy-based Security Testing Tools
- Choosing between SAST and DAST

Module 10: Secure Deployment and Maintenance

- Secure Deployment
- Pre-deployment Activities
- Deployment Activities: Ensuring Security at Various Levels
- Ensuring Security at Host Level
- Ensuring Security at Network Level
- Ensuring Security at Application Level
- Ensuring Security at Web Container Level (Tomcat)
- Ensuring Security in Oracle
- Security Maintenance and Monitoring

Documentation

- Digital course materials included

Exam

- This course prepares Certified Application Security Engineer certification
- In order to receive the Certified Application Security Engineer certification, learners must pass two exams: 312-96.
- Number of questions: 50
- Duration: 2 hours
- If you wish to take this exam, please contact our secretariat who will let you know the cost of the exam and will take care of all the necessary administrative procedures for you.

Participant profiles

- Java developers
- Application security engineers
- Software architects
- Security analysts
- Application security testers

Prerequisites

- Basic knowledge of Java development
- Understanding of IT security concepts
- Practical experience with the software development lifecycle (SDLC)
- Familiarity with security testing tools (SAST, DAST)
- Experience in software architecture

Objectives

- Understand SDLC security concepts
- Apply best practices for application security
- Use security testing tools like SAST and DAST
- Identify and fix common code vulnerabilities
- Design secure applications from the design phase
- Follow Java application security standards
- Implement cryptographic techniques in Java

Description
Java Certified Application Security Engineer (JAVA CASE) training
Niveau
Intermédiaire
Classroom Registration Price (CHF)
3950
Virtual Classroom Registration Price (CHF)
3800
Duration (in Days)
3
Reference
CASE