Comprehensive Application Security and Secure Coding Training Course



16 - 20 Nov 2026

Munich

5100 € (Per Person)

Ref: #CYB1120_88073

Course Introduction / Overview:

In today's digitally-driven world, the security of software applications is not an afterthought but a fundamental requirement of the development process. This course provides a comprehensive exploration of application security and secure coding practices, designed to equip developers, security professionals, and IT staff with the knowledge to build resilient and secure software from the ground up. The curriculum moves beyond theoretical concepts to offer practical, actionable strategies for integrating security into every phase of the software development lifecycle (SDLC). As articulated by security pioneer Gary McGraw in his influential book "Software Security: Building Security In," true software security is about proactive design and implementation, not reactive patching. This principle is the cornerstone of our training. Participants will delve into critical topics such as threat modeling, identifying and mitigating the OWASP Top 10 vulnerabilities, and implementing robust authentication and access control mechanisms. BIG BEN Training Center has structured this program to foster a security-first mindset, enabling teams to reduce vulnerabilities, minimize business risk, and deliver trustworthy applications to their users. This is an A-to-Z journey into the world of building secure and dependable software systems.

Target Audience / This training course is suitable for:

- Software Developers and Engineers.
- DevOps and DevSecOps Professionals.
- Application Security Analysts.
- Cybersecurity Consultants.
- Quality Assurance and Testing Professionals.
- IT Project Managers and Team Leads.
- System Architects and Designers.
- Penetration Testers and Ethical Hackers.

Target Sectors and Industries:

- Financial Services and Banking.
- Healthcare and Medical Technology.
- E-commerce and Retail.
- Information Technology and Software Services.
- Telecommunications.
- Government Agencies and Public Sector Organizations.
- Energy and Utilities.

Target Organizations Departments:

- Software Development and Engineering.
- Information Technology (IT).
- Cybersecurity and Information Security.
- Quality Assurance (QA) and Testing.
- Product Management.
- Research and Development (R&D).
- Internal Audit and Compliance.

Course Offerings:

By the end of this course, the participants will have able to:

- Integrate security principles throughout the entire Software Development Lifecycle (SDLC).
- Identify, analyze, and mitigate the most common web application vulnerabilities, including the OWASP Top 10.
- Apply secure coding techniques to prevent common flaws like SQL injection, Cross-Site Scripting (XSS), and security misconfigurations.
- Conduct effective threat modeling to proactively identify and address potential security risks in application design.
- Implement secure authentication, session management, and access control mechanisms.
- Understand the principles of applied cryptography and its correct usage in applications.
- Utilize various security testing methodologies, including SAST, DAST, and IAST, to discover vulnerabilities.
- Develop a security-first mindset to build more resilient and trustworthy software applications.

Course Methodology:

The training methodology at BIG BEN Training Center is designed to be immersive, practical, and highly interactive, ensuring that participants can immediately apply what they learn. We believe that mastering application security requires more than just listening to lectures; it demands hands-on engagement. The course is built around a blend of expert-led instruction, real-world case studies of security breaches, and collaborative group exercises. Participants will work in teams to perform threat modeling on sample applications, conduct secure code reviews, and develop mitigation strategies for complex vulnerabilities. Interactive sessions, Q&A panels, and peer-to-peer discussions are integrated throughout the five days to encourage knowledge sharing and critical thinking. Our approach emphasizes a problem-solving framework where participants analyze security challenges and architect robust solutions. This active learning environment, supported by continuous feedback from the instructor, ensures a deep and lasting understanding of secure coding principles and their practical implementation in a corporate setting.

Course Agenda (Course Units):

Unit One: Foundations of Application Security

- Introduction to Application Security (AppSec).
- The Importance of a Secure Software Development Lifecycle (SDLC).
- Understanding Common Security Threats and Vulnerabilities.
- Deep Dive into the OWASP Top 10: Injection and Broken Authentication.
- Principles of Secure Design and Architecture.
- Security Requirements and Risk Management.
- Introduction to Threat Modeling Concepts.

Unit Two: Secure Design and Threat Modeling

- Secure Design Principles: Least Privilege, Defense in Depth, and Fail-Secure.
- Attack Surface Analysis and Reduction.
- Threat Modeling Methodologies: STRIDE and DREAD.
- Creating Data Flow Diagrams for Security Analysis.
- Identifying Threats and Potential Vulnerabilities in Design.
- Developing Mitigation Strategies for Design-Level Flaws.
- Case Study: Threat Modeling a Real-World Application.

Unit Three: Secure Coding and Defensive Programming

- Preventing Injection Flaws: SQL, NoSQL, OS, and LDAP Injection.
- Mitigating Cross-Site Scripting (XSS): Stored, Reflected, and DOM-based.
- Secure Input Validation and Output Encoding Practices.
- Memory Management and Preventing Buffer Overflows.
- Secure Error Handling and Logging.
- Understanding and Preventing Insecure Deserialization.
- File and Resource Management Security.

Unit Four: Authentication, Authorization, and Cryptography

- Implementing Secure Authentication and Password Management.
- Best Practices for Session Management and Protection.
- Broken Access Control: Vertical and Horizontal Privilege Escalation.
- Introduction to Applied Cryptography and Its Pitfalls.
- Securely Using Cryptographic APIs for Data in Transit and at Rest.
- Managing Secrets: API Keys, Passwords, and Certificates.
- Understanding XML External Entities (XXE) and Server-Side Request Forgery (SSRF).

Unit Five: Security Testing and DevSecOps

- Static Application Security Testing (SAST) Tools and Techniques.
- Dynamic Application Security Testing (DAST) for Running Applications.
- Interactive Application Security Testing (IAST) and RASP.
- Software Composition Analysis (SCA) for Third-Party Component Security.
- Introduction to Penetration Testing and Secure Code Review.
- Integrating Security into CI/CD Pipelines (DevSecOps).
- Security Monitoring, Logging, and Incident Response Preparedness.

FAQ:

Qualifications required for registering to this course?

There are no requirements.

How long is each daily session, and what is the total number of training hours for the course?

This training course spans five days, with daily sessions ranging between 4 to 5 hours, including breaks and interactive activities, bringing the total duration to 20 - 25 training hours.

Something to think about:

What unique qualities does this course offer compared to other courses?

Beyond following a checklist of secure coding practices, how can an organization cultivate a true security-first mindset within its development teams?

What unique qualities does this course offer compared to other courses?

This training course distinguishes itself by adopting a holistic and proactive security paradigm, moving beyond the conventional, reactive approach of merely identifying and fixing bugs. Its core philosophy is centered on integrating security into the very fabric of the software development lifecycle, from initial design to final deployment. Unlike courses that focus narrowly on specific tools or vulnerabilities, this program emphasizes the cultivation of a security-first mindset. It achieves this by dedicating significant time to foundational concepts like secure design principles and threat modeling, enabling participants to anticipate and neutralize threats before a single line of code is written. The curriculum is uniquely structured to bridge the gap between development, operations, and security teams, fostering the collaborative culture essential for a successful DevSecOps implementation. Furthermore, the course content is deeply rooted in practical application, using intricate case studies and collaborative problem-solving sessions that mirror real-world challenges. The focus is not just on the "what" of application security, but the "why" and "how," ensuring participants leave with a strategic understanding and the critical thinking skills needed to build genuinely resilient software systems.