Application Security for Developers Training

Duration: **2 Day**

BİLGİNÇ IT Academy
info@bilginc.com
bilginc.com

Overview

The future of secure software depends on developers who **build security into their code**.
This hands-on {*training} provides developers, architects, and tech leads with the skills to identify, exploit, and remediate vulnerabilities.

Learners will use the **STRIDE threat modelling framework**, explore **OWASP Top 10 vulnerabilities**, and strengthen their ability to apply secure coding and defensive programming techniques.

Regulatory Compliance (BDDK)

This course **fully complies** with the
Regulation on Banks' Information Systems and Electronic Banking Services (Articles 20, 22, 23, and 25).

Covered compliance topics include:

- Secure software development and version control.
- Change and release management in DevOps pipelines.
- Vulnerability scanning and testing integration.
- Developer awareness and secure coding governance.

Recommended for **banks, insurance firms, and regulated institutions** aiming to align with BDDK mandates.

Who Should Attend

Developers, software architects, DevSecOps engineers,
and IT professionals working in **banking, finance, and regulated environments**.

Application Security for Developers Training

Duration: **2 Day**

BİLGİNÇ IT Academy
info@bilginc.com
bilginc.com

What You Will Learn

By the end of this {*training}, you will have gained knowledge and skills in the following areas:

- Apply secure development practices throughout SDLC.
- Use STRIDE threat modelling to assess application risks.
- Identify and fix vulnerabilities hands-on.
- Implement encryption and secure key management.
- Secure authentication, sessions, and APIs.
- Defend against injection, deserialization, and XSS attacks.
- Integrate security controls into Agile and DevOps workflows.
- Build a culture of security awareness across teams.

After completing this course, participants can confidently:

- Build **secure and compliant applications**,
- Integrate **security within CI/CD pipelines**,
- Meet **BDDK secure development and testing obligations**.

Outline

Application security fundamentals

- Why secure development is essential in modern software environments.
- The cost of insecure code and lessons from real-world breaches.
- Understanding the OWASP Top 10 and common developer pitfalls.
- Core threat modelling concepts and the STRIDE framework.

Developer environment security

- Protecting code in repositories and managing secure commits.
- Securing third-party dependencies and libraries.
- Automated code scanning and continuous integration security.
- Simulated attacks: phishing and supply chain compromises.

Front-end security

- Understanding the HTTP/HTTPS protocol and browser request flows.
- Identifying attack surfaces in client-side code.

BİLGİNÇ IT Academy

info@bilginc.com

bilginc.com

- Securing forms, input validation, and browser sessions.
- Applying and testing client-side security headers.
- Attacks and mitigations:
  - Cross-site scripting (XSS)
  - File upload vulnerabilities and client-side code injection
  - Session hijacking and cookie manipulation

Backend and API security

- Securing authentication and authorisation mechanisms.
- Applying secure design principles to APIs and backend logic.
- ORM and model-layer security to prevent injection and mass assignment.
- Integration security for third-party APIs and external services.
- Attacks and mitigations:
  - Brute force and login bypass
  - Parameter tampering
  - Server-side URL manipulation

Data security

- Principles of protecting data at rest and in transit.
- Implementing encryption, hashing, and key management securely.
- Understanding cryptographic vulnerabilities.
- Attacks and mitigations:
  - SQL injection
  - Insecure deserialisation

Secure file handling

- Validating file uploads and managing MIME types.
- Safely processing and storing user-uploaded documents.
- Attacks and mitigations:
  - Remote code execution via malicious uploads
  - XML external entity (XXE) attacks
  - Insecure direct object reference (IDOR)

Source code review and exploit chaining

- Conducting secure source code reviews.
- Analysing vulnerable code snippets to identify exploit chains.
- Capture the flag exercise: identifying flaws under timed conditions.

Threat modelling and agile security integration

- Applying threat modelling to full applications and incremental features.
- Building and maintaining threat lists within Agile workflows.
- Integrating security requirements into backlogs and sprints.
- Driving a team-wide security culture through process and awareness.

Exams and assessments

There are no formal exams in this course. Instead, learners complete interactive labs, practical challenges, and a competitive capture the flag activity to test their skills. Knowledge checks and guided discussions ensure participants can apply their learning to real-world projects.

Application Security for Developers Training

Duration: **2 Day**

BILGINÇ IT Academy
info@bilginc.com
bilginc.com

Hands-on learning

This course includes extensive hands-on activities, including:

- Practical threat modelling of real application features.
- Exploiting and remediating more than ten common vulnerabilities using professional security tools.
- Reviewing and securing insecure code in sandboxed environments.
- Simulated red-team exercises led by experienced penetration testers.
- A final capture the flag challenge to reinforce and test learning outcomes.