Secure Software Development Training

Duration: **3 Day**

BİLGİNÇ IT Academy
info@bilginc.com
bilginc.com

Overview

One of the most significant challenges in modern information security is software developed without sufficient consideration for security. Strong infrastructure controls alone cannot provide complete protection when the applications running within that environment contain exploitable weaknesses.

Development teams are often under pressure to deliver new functionality quickly, which can result in security requirements receiving less attention than performance, features, and delivery deadlines. This can introduce weaknesses such as insecure authentication, session management problems, injection vulnerabilities, XSS, broken access control, and unsafe data handling.

The **Secure Software Development** course is designed for software developers and security professionals who need the practical knowledge and mindset required to build applications that are more secure, controlled, and resilient.

The programme combines security concepts with practical code examples for **Java and .NET** environments. Participants examine common software vulnerabilities, understand why they occur, and learn coding and design approaches that can reduce the likelihood of exploitation.

The course covers secure input handling, authentication, session management, authorisation, secure design, code analysis, and the protection of web services and AJAX-based applications.

Prerequisites

Participants should have basic programming knowledge.

Familiarity with the following areas will be useful:

- Fundamental programming concepts
- Basic web application architecture
- HTTP request and response concepts
- Basic Java or .NET development

Secure Software Development Training

Duration: **3 Day**

BİLGİNÇ IT Academy
info@bilginc.com
bilginc.com

Who Should Attend

This course is particularly suitable for:

- Enterprise software developers
- Java developers
- .NET developers
- Application Security professionals
- Software Security specialists
- Professionals responsible for secure code reviews
- Development team leads
- QA and testing professionals
- DevSecOps practitioners
- Anyone involved in improving software security

What You Will Learn

By the end of the course, participants will be able to:

- Explain the fundamental principles of secure coding
- Identify common application security vulnerabilities
- Validate user-controlled input more safely
- Understand injection attacks and relevant prevention methods
- Identify and mitigate Cross Site Scripting vulnerabilities
- Design safer file-upload functionality
- Apply stronger authentication approaches
- Evaluate Multi-Factor Authentication options
- Reduce the risk of brute-force attacks
- Explain and mitigate session hijacking and CSRF attacks
- Apply secure session management techniques
- Protect sensitive application parameters
- Explain static and dynamic code analysis approaches
- Evaluate the role of security analysis tools
- Integrate security controls into Java and .NET development practices

Outline

Secure Coding Fundamentals

This section establishes the principles behind secure software development.

Topics include:

- Software security fundamentals
- Secure development concepts
- Core security terminology
- Confidentiality, Integrity, and Availability
- Threats, vulnerabilities, and risk

BİLGİNÇ IT Academy

info@bilginc.com

bilginc.com

- Application attack surface
- Defense in Depth
- Least Privilege
- Secure Defaults
- Integrating security into the software development lifecycle
- Java SE security fundamentals

Participants learn why application security should be addressed throughout design and development rather than added only during final testing.

Secure Input Handling

This section examines the risks associated with accepting data from users and external systems.

Topics include:

- Input validation
- Buffer Overflow
- Cross Site Scripting
- XSS prevention
- SQL Injection
- XML Injection
- LDAP Injection
- Injection prevention techniques
- Arbitrary File Uploads
- Secure file upload
- Secure image upload
- Canonicalization
- Normalization
- Regular Expressions
- Allowlisting / whitelisting
- Blocklisting / blacklisting

Participants apply the principle that externally controlled data should not be trusted and should be validated appropriately on the server side.

Cross Site Scripting and XSS Prevention

This section looks more closely at why XSS vulnerabilities occur and how they can be reduced.

Topics include:

- Understanding XSS
- Reflected XSS
- Stored XSS
- Safely handling browser output
- Input validation
- Output encoding
- Context-aware encoding
- XSS prevention techniques

Injection Attacks

BİLGİNÇ IT Academy

info@bilginc.com

bilginc.com

This section examines attacks caused by untrusted data being passed into backend interpreters and services.

Topics include:

- SQL Injection
- XML Injection
- LDAP Injection
- Injection attack behaviour
- Parameterized queries
- Safe API use
- Input validation
- Injection prevention

Secure File Uploads

File-upload functionality can create a significant application attack surface if it is not carefully controlled.

Topics include:

- Arbitrary File Upload
- File extension validation
- MIME type validation
- Content validation
- Secure image uploads
- Secure upload locations
- Filename validation
- Canonicalization risks
- Preventing malicious file uploads

Secure Output Handling

This section focuses on safely presenting data back to users.

Topics include:

- Output encoding
- Anti-XSS output encoding
- HTML context
- Attribute context
- JavaScript context
- URL context
- Safely rendering user-controlled content

Participants learn why the correct encoding approach depends on the context in which the data is displayed.

Protecting Sensitive Parameters

This section focuses on the protection of sensitive information used by applications.

Topics include:

BİLGİNÇ IT Academy

info@bilginc.com

bilginc.com

- Sensitive parameters
- Passwords
- Tokens
- API keys
- Session identifiers
- Sensitive information in query strings
- Sensitive information in logs
- Secure storage
- Secrets management
- Minimising unnecessary exposure of sensitive data

Secure Authentication

This module examines authentication mechanisms and their security implications.

Topics include:

- Authentication concepts
- Authentication types
- Basic Authentication
- Digest Authentication
- Windows Authentication
- Forms Authentication
- Secure authentication methods
- Credential protection
- Password security
- Multi-Factor Authentication
- Brute-force attacks
- Account lockout
- Rate limiting
- Brute-force prevention algorithms

Participants examine common authentication weaknesses and controls that can reduce associated risks.

Secure Session Management

This module focuses on protecting user sessions throughout their lifecycle.

Topics include:

- Session management
- Session attacks
- Session fixation
- Session hijacking
- Secure session identifiers
- Session timeouts
- Logout handling
- Cookie security
- HttpOnly
- Secure flag
- Java Servlet Filters
- Session invalidation

BİLGİNÇ IT Academy

info@bilginc.com

bilginc.com

Cross Site Request Forgery

Topics include:

- Understanding CSRF
- CSRF attack flow
- State-changing requests
- Anti-CSRF tokens
- SameSite cookies
- CSRF prevention approaches
- Request validation
- Relationship between sessions and CSRF

Secure Authorisation

The course also examines how access to functions and resources should be controlled after a user has authenticated.

Topics include:

- Authentication versus Authorisation
- Access control
- Role-Based Access Control
- Privilege validation
- Server-side authorisation
- Least Privilege
- Broken access control
- Function-level access control
- Object-level authorisation

Secure Design

Participants examine security weaknesses caused by architecture and design choices rather than coding mistakes alone.

Topics include:

- Secure by Design
- Threat modelling
- Attack surface reduction
- Trust boundaries
- Defense in Depth
- Fail Secure
- Least Privilege
- Secure architecture decisions
- Integrating security requirements into design

Secure Code Analysis

This section examines how security issues can be identified during development and testing.

Topics include:

BİLGİNÇ IT Academy

info@bilginc.com

bilginc.com

- Secure code review
- Static Application Security Testing
- Dynamic Application Security Testing
- Static code analysis
- Dynamic analysis
- False positives and false negatives
- Code analysis tools
- Evaluating security findings
- Prioritising vulnerabilities
- Validating remediation

Secure Web Services and AJAX

This section focuses on security considerations for APIs, web services, and asynchronous web applications.

Topics include:

- Web service security
- API authentication
- API authorisation
- Input validation
- JSON and XML security
- AJAX security
- Sensitive data exposure
- Cross-origin risks
- Secure error handling
- Session and token security for web services