

C# and Web Application Security Eğitimi

Süre: 3 Gün

Açıklama

C# Web Uygulama Güvenliği Eğitimi, C#, ASP.NET güvenliği ve güvenli kodlama konularını ele alan 3 günlük bir eğitim programıdır.

ASP.NET uygulamalarını etkileyen web güvenlik açıkları, .NET güvenlik mimarisi, kriptografi ve sık karşılaşılan kodlama hataları canlı uygulamalı alıştırma ile ele alınır.

Bu eğitim, C# kodlarına bakışınızı kökünden değiştirecek. Siber saldırganların sisteminize saldırabilmelerini sağlayan genel zayıflıklar ve bunların yol açtıkları sonuçların yanında daha da önemlisi kendinizi koruyabilmenizi sağlayan en iyi uygulamalar hakkında bilgi sahibi olacaksınız. Bu eğitimde .Net çerçevesinin kriptografi veya Kod Erişim Güvenliğinin kullanımı gibi çeşitli güvenlik yönleri hakkında bütünsel bir bakış açısının yanında aynı zamanda kaçınmanız gereken çok sık yapılan C# programlama hataları hakkında bilgi sahibi olacaksınız. Ayrıca CLR'den modern AJAX ve HTML5 tabanlı ön uçlara kadar tüm yığında, ASP.NET Web uygulamalarını nasıl etkilediklerine odaklanan özgün Web güvenlik açıkları da ele alınıyor. Eğitim, daha eğlenceli ve çekici hale gelmesi için canlı uygulamalı alıştırma eşliğinde gerçekleştirilir.

Bu eğitimde neler öğreneceksiniz?

- Güvenlik, BT güvenliği ve güvenli kodlamanın temel kavramları
- OWASP Top Ten sıralamasının ötesindeki Web güvenlik boşlukları ve bunlardan nasıl kaçınılacağı
- XML güvenliği hakkında bilgiler
- Dağıtım ortamının nasıl güvenli bir şekilde kurulacağı ve işletileceği
- İstemci tarafı güvenlik boşlukları ve güvenli kodlama uygulamaları
- .Net geliştirme ortamının çeşitli güvenlik özelliklerini kullanma
- Kriptografi hakkında pratik bilgiler
- Tipik kodlama hataları ve bu hatalardan kaçınma yöntemleri
- Güvenli kodlama uygulamaları hakkında çeşitli kaynaklar ve diğer bilgiler

Eğitim İçeriği

IT security and secure coding

- Nature of security
- What is risk?
- IT security vs. secure coding
- From vulnerabilities to botnets and cybercrime
- Nature of security flaws
- Reasons of difficulty
- From an infected computer to targeted attacks
- The Seven Pernicious Kingdoms
- OWASP Top Ten 2017

Web application security

- Injection
- Injection principles
- SQL injection
- Exercise – SQL Injection
- Exercise – SQL injection
- Typical SQL Injection attack methods
- Blind and time-based SQL injection
- SQL injection protection methods
- Other injection flaws
- Command injection
- Command injection exercise – starting Netcat
- Case study – ImageMagick
- Cookie injection / HTTP parameter pollution
- Exercise – Value shadowing
- Broken authentication
- Session handling threats
- Session fixation
- Exercise – Session fixation
- Session handling best practices
- Setting cookie attributes – best practices
- Sensitive data exposure
- Transport layer security
- XML external entity (XXE)
- XML Entity introduction
- XML bomb
- Exercise – XML bomb
- XML external entity attack (XXE) – resource inclusion

- XML external entity attack – URL invocation
- XML external entity attack – parameter entities
- Exercise – XXE attack
- Preventing entity-related attacks
- Case study – XXE in Google Toolbar
- Broken access control
- Typical access control weaknesses
- Insecure direct object reference (IDOR)
- Exercise – Insecure direct object reference
- Protection against IDOR
- Case study – Facebook Notes
- Failure to restrict URL access
- Security misconfiguration
- Configuration management
- Hardening
- Patch management
- NET components and environment overview
- Insecure file uploads
- Exercise – Uploading executable files
- Filtering file uploads – validation and configuration
- Cross-Site Scripting (XSS)
- Persistent XSS
- Reflected XSS
- DOM-based XSS
- Exercise – Cross Site Scripting
- Exploitation: CSS injection
- Exploitation: injecting the <base> tag
- Exercise – HTML injection with base tag
- XSS prevention
- Output encoding API in C#
- XSS protection in ASP.NET – validateRequest
- Web Protection Library (WPL)

Web application security

- Insecure deserialization
- Deserialization basics
- Security challenges of deserialization
- Deserialization in .NET

- From deserialization to code execution
- POP payload targeting MulticastDelegate (C#)
- Real-world .NET examples of deserialization vulnerabilities
- Issues with deserialization – JSON
- Best practices against deserialization vulnerabilities
- Using components with known vulnerabilities
- Vulnerability attributes
- Common Vulnerability Scoring System – CVSS

Client-side security

- JavaScript security
- Same Origin Policy
- Cross Origin Resource Sharing (CORS)
- Exercise – Client-side authentication
- Client-side authentication and password management
- Protecting JavaScript code
- Exercise – JavaScript obfuscation
- Clickjacking
- Exercise – Do you Like me?
- Protection against Clickjacking
- Anti frame-busting – dismissing protection scripts
- Protection against busting frame busting
- AJAX security
- XSS in AJAX
- Script injection attack in AJAX
- Exercise – XSS in AJAX
- XSS protection in Ajax
- Exercise CSRF in AJAX – JavaScript hijacking
- CSRF protection in AJAX
- HTML5 security
- New XSS possibilities in HTML5
- HTML5 clickjacking attack – text field injection
- HTML5 clickjacking – content extraction
- Form tampering
- Exercise – Form tampering
- Cross-origin requests
- HTML proxy with cross-origin request
- Exercise – Client side include

.NET security architecture and services

- .NET architecture
- Code Access Security
- Full and partial trust
- Evidence classes
- Permissions
- Code access permission classes
- Deriving permissions from evidence
- Defining custom permissions
- .NET runtime permission checking
- The Stack Walk
- Effects of Assert()
- Class and method-level declarative permission
- Imperative (programmatic) permission checking
- Exercise – sandboxing .NET code
- Using transparency attributes
- Allow partially trusted callers
- Exercise – using transparency attributes

Practical cryptography

- Cryptosystems
- Elements of a cryptosystem
- Symmetric-key cryptography
- Providing confidentiality with symmetric cryptography
- Symmetric encryption algorithms
- Block ciphers – modes of operation
- Other cryptographic algorithms
- Hash or message digest

Ön Koşullar

C# Programlama tecrübesi