Güncel teknolojiler artık dünya ile iletişimimiz daha hızlı ve farklı bir boyuta taşımaktadır. Bu etkileşim içinde olunan dünyada ise özellikle web uygulamaları 3/4 oranında kullanılıyor. Farklı şekillerde karşımıza çıkan bu kritik uygulamalar, bankacılıktan, devlet ve kamusal uygulamalara kadar birçok sektör ve alanı kapsamaktadır. Hal böyle olduğunda performansın yanı sıra güvenlik de ön plana çıkan bir unsur halini alıyor. Vakitten ve nakitten kazanılan bu uygulamalarda güvenlik sistemleri ise ayrı bir alan olarak ele alınıyor. **Güvenli Kodu Geliştirme** eğitimleri de sektörün en önde gelen programcılıkları içinde yerini almış durumda. Verilerin ve insanların bilgi güvenliğini en üst seviyede koruyabilmek amacı ile geliştirilmiş olan bu uygulamalar, farklı beklentilerin de en etkin şekilde karşılanmasını sağlıyor. Sizler de bu özel ve önü açık olan eğitimi kendiniz ya da kariyerinizi şekillendirmek adına alabilirsiniz. Bu şekilde güncel bir yazılım ve kodlama sistemine ait tüm bilgileri de edinmiş olacaksınız.
Yazılımların güvenli olması kavramsal olarak artık hepimizi ilgilendiren bir konu. Yaptığınız basit bir işlemden, en büyüğüne kadar her şekilde sizler de güvenli işlemler yapmayı hedeflersiniz. Bu her kullanıcı için aynı şekilde geçerlidir. Bu eğitimler sonucunda, yazılımlarda sırasında çıkacak olan güvenlik açıklarını da anında kapatabilir, etkileyici sonuçları elde edebilirsiniz. Kaynakta müdahale her bakımdan bu tip işlemlerde en etkin sonucu sağlayacaktır. Kurumumuz sunmuş olduğu **Güvenlik Kodu Geliştirme** eğitimleri ile yazılımlarda ki bu dönüşümü elde edebilmeniz amacı ile sizleri tam bir donanıma sokacaktır. Var olan yazılımlar ya da yeni yaratacağınız tüm yazılımlara daha haiz olarak güvenliği en uygun şekilde kodlayabileceksiniz. Bu uygulama aynı zamanda yük testlerini yapabilmenize ve performansların da aynı şekilde yükselmesine olanak sağlayacaktır. Güvenliğin her alanda olduğu gibi bilişim sistemi içinde de gerçekleşmesi oldukça özel bir durumdur. Bu neden ile yapılacak olan her işlem hayati önem taşımaktadır.
Her türlü kod gelişiminde güvenliğin önemi oldukça büyüktür. Bu neden ile yazılımlarını korumak, elde edeceğiniz performansın denetlenmesi, farklı şekillerde çıkacak olan güvenlik açıklarının kaynağında kapatılması amacı ile bu tarz bir eğitim ile en yüksek performans sağlanmış olacaktır. Sizlere sunulan bu harika ve özel eğitim için belirli bir alt yapınızın olması da gerekmektedir. İlerleyen aşamalarda ise **Güvenlik Kodu Geliştirme** eğitimi üzerine yeni ve farklı eğitimleri ilave ederek daha farklı bir alanda kariyerinize de devam edebilirsiniz. Etkin şekilde kullanıma uygun olan bu kodlama sistemi, bilişim ve verilerin de en doğru şekilde korunmasında büyük önem taşımaktadır. Sizler de farklı şekillerde yaptığınız uygulamalar, web sayfaları ve diğer uygulamalar için kesinlikle yüksek güvenlik önemlerini almak isteyeceksiniz. Bu neden ile kurumumuzun sizlere özel sunduğu bilişime yönelik güvenlik kod eğitimlerine her şekilde katılabilirsiniz.

Eğitim Süresi: 3 Gün
İçerik

Introduction To Defensive Programming

Anatomy of an Attack
Risks and Rewards
Building Security from the Ground Up
Defense in Depth
Never Trust Input
Fail Gracefully
Watch for Attacks
Use Least Privilege
Firewalls and Cryptography Are Not a Panacea
Security Should Be Your Default State
Code Defensively
The OWASP Top Ten
Moving Forward
Checklists

.Net Security Best Practices(Writing Secure .Net Codes)

Layer Based Coding Techniques (Where to Handle Input Validation? Etc.)

Asp.Net Security Best Practices

How the Web Works
Examining HTTP
Requesting a Resource
Responding to a Request
Sniffi ng HTTP Requests and Responses
Understanding HTML Forms
Examining How ASP.NET Works
Understanding How ASP.NET Events Work
Examining the ASP.NET Pipeline
Writing HTTP Modules

Safely Accepting User Input

Defining Input
Dealing with Input Safely
Echoing User Input Safely
Mitigating Against XSS
The Microsoft Anti-XSS Library
The Security Run-time Engine
Constraining Input
Protecting Cookies
Validating Form Input
Validation Controls
Standard ASP.NET Validation Controls
A Checklist for Handling Input

Using Query Strings, Form Fields, Events,and Browser Information

Using the Right Input Type
Query Strings
Form Fields
Request Forgery and How to Avoid It
Mitigating Against CSRF
Protecting ASP.NET Events
Avoiding Mistakes with Browser Information
A Checklist for Query Strings, Forms, Events,and Browser Information

Controlling Information

Controlling ViewState
Validating ViewState
Encrypting ViewState
Protecting Against ViewState One-Click Attacks
Removing ViewState from the Client Page Disabling Browser Caching
Error Handling and Logging
Improving Your Error Handling
Watching for Special Exceptions
Logging Errors and Monitoring Your Application
Using the Windows Event Log
Using Email to Log Events
Using ASP.NET Tracing
Using Performance Counters
Using WMI Events

Another Alternative: Logging Frameworks
Limiting Search Engines
Controlling Robots with a Metatag
Controlling Robots with robots.txt
Protecting Passwords in Confi g FilesA Checklist for Query Strings, Forms, Events, and

Authorization and Authentication

Discovering Your Own Identity
Adding Authentication in ASP.NET
Using Forms Authentication
Confi guring Forms Authentication
Using SQL as a Membership Store
Creating Users
Examining How Users Are Stored
Confi guring the Membership Settings
Creating Users Programmatically
Supporting Password Changes and Resets
Windows Authentication
Confi guring IIS for Windows Authentication
Impersonation with Windows Authentication
Authorization in ASP.NET
Examining <allow> and <deny>
Role-Based Authorization
Confi guring Roles with Forms-Based Authentication
Using the Confi guration Tools to Manage Roles
Managing Roles Programmatically
Managing Role Members Programmatically
Roles with Windows Authentication
Limiting Access to Files and Folders
Checking Users and Roles Programmatically
Securing Object References
A Checklist for Authentication and Authorization

Hashing and Encrypton

Protecting Integrity with Hashing
Choosing a Hashing Algorithm
Protecting Passwords with Hashing
Salting Passwords
Generating Secure Random Numbers
Encrypting Data
Understanding Symmetric Encryption
Protecting Data with Symmetric Encryption
Sharing Secrets with Asymmetric Encryption
Using Asymmetric Encryption without Certifi cates
Using Certifi cates for Asymmetric Encryption
Getting a Certifi cate
Using the Windows DPAPI
A Checklist for Encryption

Securely Accessing Databases

Writing Bad Code: Demonstrating SQL Injection
Fixing the Vulnerability

More Security for SQL Server
Connecting Without Passwords
SQL Permissions
Adding a User to a Database
Managing SQL Permissions
Groups and Roles
Least Privilege Accounts
Using Views
SQL Express User Instances
Drawbacks of the VS Built-in Web Server
Dynamic SQL Stored Procedures
Using SQL Encryption
Encrypting by Pass Phrase
SQL Symmetric Encryption
SQL Asymmetric Encryption
Calculating Hashes and HMACs in SQL
A Checklist for Securely Accessing Databases

Using the File System

Accessing Existing Files Safely
Making Static Files Secure
Checking That Your Application Can Access Files
Making a File Downloadable and Setting Its Name
Adding Further Checks to File Access
Adding Role Checks
Anti-Leeching Checks
Accessing Files on a Remote System
Creating Files Safely
Handling User Uploads
Using the File Upload Control
A Checklist for Securely Accessing Files

Securing XML

Validating XML
Well-Formed XML
Valid XML
XML Parsers
Querying XML
Avoiding XPath Injection
Securing XML Documents
Encrypting XML Documents
Using a Symmetric Encryption Key with XML
Using an Asymmetric Key Pair to Encrypt and Decrypt XML
Using an X509 Certificate to Encrypt and Decrypt XML
Signing XML Documents
A Checklist for XML

Windows Communication Foundation Security

Creating and Consuming WCF Services
Security and Privacy with WCF
Transport Security
Message Security

Mixed Mode
Selecting the Security Mode
Choosing the Client Credentials
Adding Security to an Internet Service
Signing Messages with WCF
Logging and Auditing in WCF
Validating Parameters Using Inspectors
Using Message Inspectors
Throwing Errors in WCF
A Checklist for Securing WCF

Understanding Code Access Security

Using ASP.NET Trust Levels
Demanding Minimum CAS Permissions
Asking and Checking for CAS Permissions
Testing Your Application Under a New Trust Level
Using the Global Assembly Cache to Run Code Under Full Trust
.NET 4 Changes for Trust and ASP.NET
A Checklist for Code not Under Full Trust

Securing Internet Information Server (IIS)

Installing and Confi guring IIS7
IIS Role Services
Removing Global Features for an Individual Web Site
Creating and Confi guring Application Pools
Confi guring Trust Levels in IIS
Locking Trust Levels
Creating Custom Trust Levels
Filtering Requests
Filtering Double-Encoded Requests
Filtering Requests with Non-ASCII Characters
Filtering Requests Based on File Extension
Filtering Requests Based on Request Size
Filtering Requests Based on HTTP Verbs
Filtering Requests Based on URL Sequences
Filtering Requests Based on Request Segments
Filtering Requests Based on a Request Header
Status Codes Returned to Denied Requests
Using Log Parser to Mine IIS Log Files
Using Certifi cates
Requesting an SSL Certifi cate
Confi guring a Site to Use HTTPS
Setting up a Test Certifi cation Authority
A Checklist for Securing Internet Information Server (IIS)

Security Testing

Performing a Security Code Review
A Designer's Security CheckList
A Developer's Security CheckList
A Tester's Security CheckList