

Implementing Security for Applications with Microsoft C# .NET

- **Course Number:** 2840 C#
- **Length:** 5 Day(s)

Certification Exam

This class will help the student prepare for the following Microsoft Certification exams:

- **Exam 70-340:** Implementing Security for Applications with Microsoft C# .NET

Course Overview

This five-day instructor-led class provides students with a thorough grounding in Microsoft .NET security implementation and general development security best practices. This course will prepare a student to take the Implementing Security for Applications exam (Microsoft Visual C# 70-340).

Prerequisites

Before attending this course, students:

- Should have a minimum of 1 year of experience using Microsoft Visual Studio .NET 2003 (.NET Framework 1.1) and 2–3 years of additional development experience.
- Should be experienced in Visual C#.

Audience

This course is intended for experienced, professional application developers, including those employed by software companies or working on corporate development teams.

Course Outline

Level 1

Secure.Net Overview

- 1.1 .Net Assemblies
- 1.2 Assembly Parts
- 1.3 Metadata
- 1.4 Strong Name and Reflection
- 1.5 Security Overview
- 1.6 Type Safety Verification
- 1.7 Code Signing
- 1.8 Encryption and Digital Signatures
- 1.9 Code Access and Role Base Security
- 1.10 Isolated Storage
- 1.11 Application Deployment
- 1.12 Versioning
- 1.13 Configuration
- 1.14 Review

Metadata and Reflection

- 2.1 Metadata
- 2.2 Demo - ILDASM.exe
- 2.3 Reflection
- 2.4 Demo - Reflection
- 2.5 Review

Lab - Using the (MSIL) Disassembly

Secure Coding

- 3.1 Security Basics
- 3.2 Security Measures

- 3.3 Malicious Users & .Net
- 3.4 Best Practices
- 3.5 Threat Modeling
- 3.6 Creating Threat Models
- 3.7 Applying Threat Models
- 3.8 Review

Level 2

Cryptography

- 1.1 Cryptography and Digital Signing
- 1.2 Symmetric and Asymmetric Scenarios
- 1.3 Cryptography in the .Net Framework
- 1.4 Cryptography with Symmetric Algorithms
- 1.5 Demo - Symmetric Cryptography
- 1.6 Cryptography with Asymmetric Algorithms
- 1.7 Demo - Hashing
- 1.8 Signing Code
- 1.9 Demo - Strong Names
- 1.10 Review

Lab - Using the File Signing Tool

Level 3

Code Access Security

- 1.1 Evidence
- 1.2 Security Policy
- 1.3 Code Groups
- 1.4 Security Policy Level
- 1.5 Modifying Security Policy
- 1.6 Demo - Graphical Configuration
- 1.7 CasPol Tool
- 1.8 Demo CasPol Tool
- 1.9 Security Operation Basics
- 1.10 Permission Demand
- 1.11 Permission Assert
- 1.12 Other Security Checks
- 1.13 Imperative and Declarative Security
- 1.14 Allow Partially Trusted Callers Attribute
- 1.15 Imperative Security
- 1.16 Demo - Imperative Security
- 1.17 Declarative Security
- 1.18 Demo - Declarative Security
- 1.19 Review

Lab - Administrating Security Policy

Level 4

Role Based Security

- 1.1 Creating Windows Principal and Identity
- 1.2 Demo - Principal and Identity
- 1.3 Generic Identity and Principal
- 1.4 Demo - Generic Authentication
- 1.5 Principal Permission Object
- 1.6 Demo - Principal Permission
- 1.7 Review

Lab - Assign Users to Security Role

Level 5

Isolated Storage

- 1.1 Defining Isolated Storage
- 1.2 Using Isolated Storage
- 1.3 Demo - Isolated Storage
- 1.4 Review

Creating and Assembly

- 2.1 Single and Multi File Assemblies
- 2.2 Demo - Command Line Compilation
- 2.3 Private VS Shared Assemblies
- 2.4 Demo - Global Assembly Cache
- 2.5 Review

Deploying .Net Applications

- 3.1 Deployment Methods
- 3.2 Creating a Setup Project
- 3.3 Demo - Deployment
- 3.4 Review

Lab - Deploying an Application

Level 6

Assembly Binding Configuration

- 1.1 Assembly Binding Basics
- 1.2 Side by Side Deployment
- 1.3 Configuration Files
- 1.4 Assembly Binding Process
- 1.5 Configuration File Syntax
- 1.6 Creating Policy Configuration Files
- 1.7 Demo - Assembly Reflection
- 1.8 Review

Lab - Binding and Configuration

Introduction to Web Security

- 2.1 Importance of Security
- 2.2 Security Challenges
- 2.3 Hackers and Attackers
- 2.4 Attack Types
- 2.5 Vulnerabilities
- 2.6 Implementing Security
- 2.7 Best Practices
- 2.8 Review

Level 7

Validating User Input

- 1.1 Type of User Input
- 1.2 Why Validate Input
- 1.3 Type of Validation
- 1.4 User Input Attacks
- 1.5 HTTP Cookie and Header Attacks
- 1.6 Form Data and Script Attacks
- 1.7 Demo -Web Form Attacks
- 1.8 Performing Validation
- 1.9 Concealing Information
- 1.10 Review

Lab - The STRIDE Threat Model

Securing Web Pages

- 2.1 ASP.Net Authentication Methods
- 2.2 Configuration ASP.Net
- 2.3 Windows Based Authentication
- 2.4 Demo -Windows Security
- 2.5 Form Based Authentication
- 2.6 Implementing Form Based Authentication
- 2.7 Demo - Forms Security
- 2.8 Review

Level 8

Server Security

- 1.1 Internet Information Services IIS
- 1.2 Impersonation and User ID
- 1.3 Configuring Permissions
- 1.4 Client Authentication
- 1.5 Application Protection Level
- 1.6 Demo - IIS
- 1.7 Windows Server 2000/2003
- 1.8 Access Control Lists
- 1.9 Windows Server Best Practices
- 1.10 Demo - Creating ACLs
- 1.11 SQL Server
- 1.12 Authentication and Permissions
- 1.13 SQL Server Best Practices
- 1.14 Demo - SQL Server
- 1.15 SQL to IIS Security
- 1.16 SQL Injection Attacks
- 1.17 Demo - Injection Attacks
- 1.18 Injection Attack Protection
- 1.19 Review

Lab - Internet Information Services

Level 9

Protecting Communication

- 1.1 Digital certificates
- 1.2 SSL/TLS

1.3 IPSec

1.4 Review

Web Applications

2.1 Web Security Difference

2.2 Creating a Test Plan

2.3 Performing a Security Test

2.4 Review

Lab - The Dread Threat Model

Best Practices

3.1 Web Service Enhancements WSE

3.2 Cryptography

3.3 Web Application Security

3.4 User Input

3.5 General Good Practices

3.6 Critical Best Practices

3.7 Review