

## Developing and Deploying Secure Microsoft .NET Framework Applications (MS2350)

**Duration:** 3 Days

### Description

This three-day instructor-led course teaches developers to develop and deploy secure Microsoft .NET client applications by using Microsoft Visual Studio .NET and the Microsoft .NET Framework. The course provides an overview of security; discusses design issues, including threat modeling techniques and coding techniques that enhance security; and explains why type-safety verification is the cornerstone of Microsoft .NET Framework security. This course provides side-by-side coding examples and activities written in both Microsoft Visual Basic .NET and Microsoft Visual C#.

### Audience

This course is intended for experienced professional software developers who work on development teams in a corporate enterprise or for independent software vendors. These developers may also develop enterprise level applications in a networked environment.

### Prerequisites

Developers who will gain the most from this course have a working understanding of the .NET Framework and some project experience writing .NET Framework client applications by using either Visual Basic .NET or Visual C#.

### Topics

#### INTRODUCTION TO .NET FRAMEWORK SECURITY AND DEPLOYMENT

This module introduces concepts and terminology, including a working definition of assembly, that are related to security and deployment in the Microsoft .NET Framework.

- Introduction to .NET Assemblies
- Overview of Security Measures
- Overview of Deployment Concepts

#### VIEWING METADATA AND USING REFLECTION

This module discusses metadata as it applies to assemblies and types. Reading metadata in Microsoft intermediate language (MSIL) code enables you to understand and troubleshoot assembly and

type references. This module also discusses techniques for programmatically accessing metadata by using reflection.

- Viewing Metadata
- Using Reflection

#### SECURE CODING AND TYPE-SAFETY VERIFICATION

This module provides an overview of security, discusses some design and coding techniques that enhance security, and then explains why type-safety verification is the cornerstone of Microsoft .NET Framework security.

- Model Type-Safety
- Verification

## Developing and Deploying Secure Microsoft .NET Framework Applications (MS2350)

### CRYPTOGRAPHY AND DIGITAL SIGNING

This module discusses cryptography and digital signing. These technologies involve the protection of data and code. You can encrypt data to prevent unauthorized users from viewing it, and you can sign both data and code to prevent tampering and to identify the sender. The Microsoft .NET Framework provides extensive support for cryptography and data signing.

- Cryptography and Signing Basics
- Encrypting and Decrypting Data with a Symmetric Algorithm
- Encrypting, Decrypting, and Signing Data with an Asymmetric Algorithm Signing Code

*Lab: Encrypting and Decrypting Text with a Password*

- Generate a key for a symmetric algorithm from a password and a random number.
- Encrypt data by using a symmetric algorithm.
- Decrypt data by using a symmetric algorithm.

### CODE ACCESS SECURITY

This module discusses code access security. This feature of the .NET Framework allows the developer and the systems administrator to exercise precise control over the resources that code is given permission to access. You can use tools and classes that are provided with the Microsoft .NET Framework to view and modify how code access security is implemented in your application.

- Overview of Code Access Security

- Modifying Security Policy
- Security Operations Basics
- Performing Imperative Security Operations
- Performing Declarative Security Operations
- Adding Permission Requests

*Lab: Using Code Access Security*

- Perform demand and assert operations by using imperative code access security
- Add minimum and optional permission requests to an assembly

### ROLE-BASED SECURITY

This module discusses programming techniques for implementing role-based security by using the Microsoft .NET Framework.

- Role-Based Security Basics
- Role-Based Security with Principal and Identity Objects
- Role-Based Security with Permission Objects

*Lab: Role-Based Security*

- Perform a role-based security check by using a principal object
- Perform a role-based security check by using a permission object
- Perform a role-based security check by using a permission attribute

### ISOLATED STORAGE

This module discusses isolated storage, what it is, the advantages of using it, and how to use it.

- Isolated Storage Basics

## Developing and Deploying Secure Microsoft .NET Framework Applications (MS2350)

- Using Isolated Storage

### CREATING AN ASSEMBLY

This module describes why and how to deploy an assembly either as a single file or as multiple files. It also describes why and how to deploy an assembly privately or as a shared assembly.

- Creating Single-File and Multifile Assemblies
- Creating Privately Deployed and Shared Assemblies

### DEPLOYING .NET FRAMEWORK APPLICATIONS

This module discusses specific reasons for using each deployment option. It also describes how to create deployment projects and how to customize deployment.

- Overview of Deployment
- Creating a Setup

### ASSEMBLY BINDING AND CONFIGURATION

This module covers how to configure assembly binding by using the Microsoft .NET Framework. The ability to manage assembly binding allows you to perform the following deployment tasks: Deployment of an updated shared component across an enterprise. Allow a specific application to continue to use an earlier version of a shared assembly. Enforce binding policy across the enterprise without exception.

- Versioning and Assembly Binding
- Basics Configuration File Syntax
- Creating Policy Configuration Files