LESSON 5

98-363 Web Development Fundamentals

5.1 Configuring Authentication, Authorization, and Impersonation

5.2 Configuring Projects, Solutions, and Reference Assemblies

5.3 Publish Web Applications

5.4 Understand Application Pools

MTA Web Development Fundamentals 5 Test

Microsoft

LESSON 5.1

98-363 Web Development Fundamentals

Configuring Authentication, Authorization, and Impersonation

Lesson Overview

In this lesson, you will review:

- The various authentication schemes that can be used in a Web application
- The difference between Authentication vs. Authorization

Authentication vs. Authorization

- Authorization
  - Accepts credentials from the user
  - Validates the credentials
- Authentication
  - Given the authentication credentials supplied, determines the right to access a resource
  - Assign authorization by user name or by role

Authentication Methods

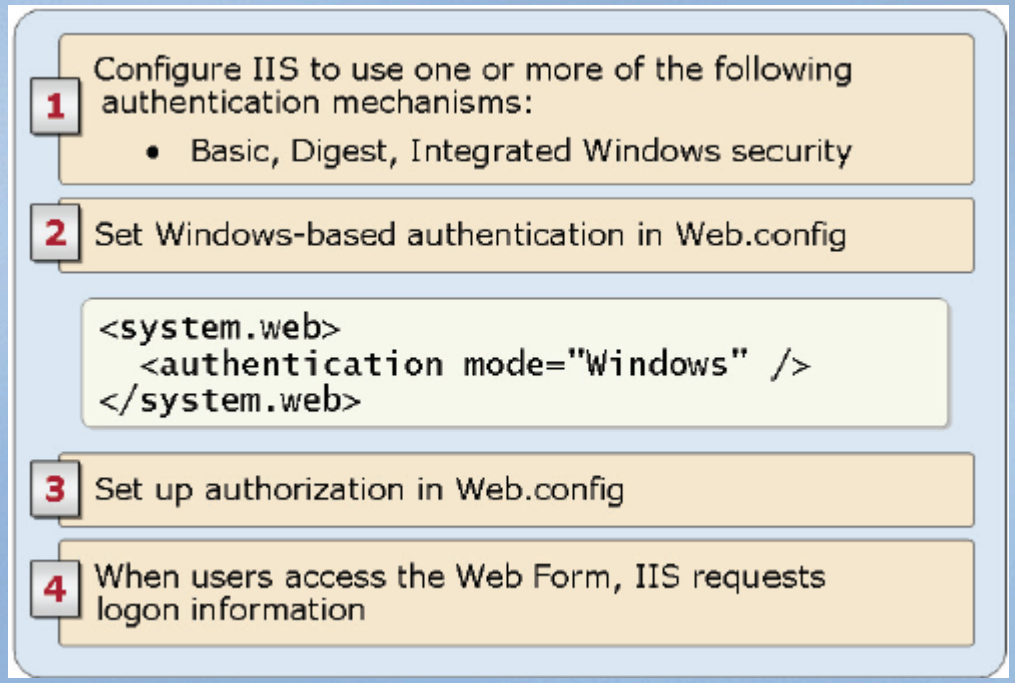
- Windows-based:
  - Relies on the Windows OS and IIS
  - User requests a secure Web page and the request goes through IIS
  - Once credentials are verified by IIS, the secure Web page is returned
- Forms-based:
  - Unauthenticated request are redirected to an HTML form
  - User provides credentials and submits the HTML form
  - After credentials are verified, an authorization cookie is issued

Comparison of the ASP.NET

Authentication Methods

Method	Advantages	Disadvantages
Windows-based Authentication	▪ Uses existing Windows infrastructure ▪ Controls access to sensitive information	▪ Not appropriate for most Internet applications
Forms-based Authentication	▪ Good for Internet applications ▪ Supports all client types	▪ Based on cookies

Enabling Windows-Based Authentication



After authentication, the Web server reads the user identity

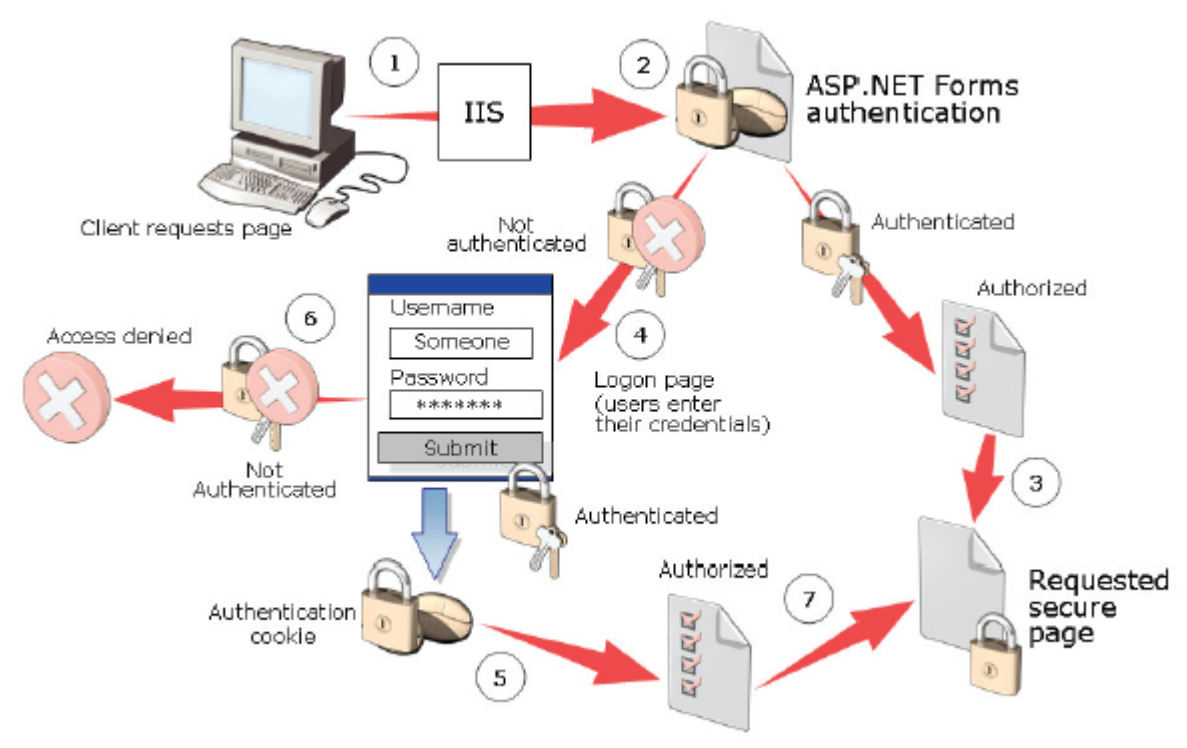
```
[Visual C#]
userLabel.Text = User.Identity.Name;
userTypeLabel.Text = User.Identity.AuthenticationType;
userAuthenticatedLabel.Text = User.Identity.IsAuthenticated;
```

```
[Visual Basic]
userLabel.Text = User.Identity.Name
userTypeLabel.Text = User.Identity.AuthenticationType
userAuthenticatedLabel.Text = User.Identity.IsAuthenticated
```

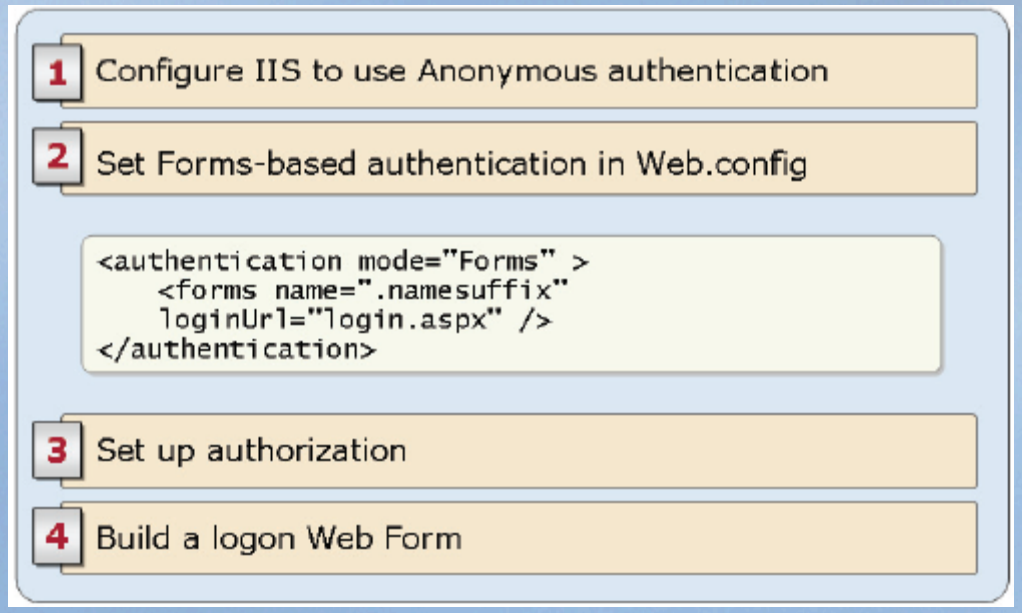
LESSON 5.1

98-363 Web Development Fundamentals

Forms-Based Authentication



How to Enable Forms-Based Authentication



Assignment

- Complete student activity 5.1

Microsoft

LESSON 5.2

98-363 Web Development Fundamentals

Configuring Projects, Solutions, and Reference Assemblies

Lesson Overview

- In this lesson you will review the following topics:
  - Local assemblies
  - Shared assemblies (GAC)
  - Web application projects
  - Configuration files

Assemblies

- The building blocks of .NET Framework applications
- Form the fundamental unit of deployment, version control, reuse, activation scoping, and security permissions
- A collection of types and resources that are built to work together and form a logical unit of functionality
- Provides the common language runtime with the information it needs to be aware of type implementations. To the runtime, a type does not exist outside the context of an assembly
- *As a general guideline, keep assembly dependencies private, and locate assemblies in the application directory unless sharing an assembly is explicitly required.*

Global Assembly Cache (GAC)

- Deploying an assembly into the global assembly cache
  - Use an installer designed to work with the global assembly cache. This is the preferred option for installing assemblies into the global assembly cache.
  - Use a developer tool called the Global Assembly Cache tool (Gacutil.exe), provided by the Windows Software Development Kit (SDK)
  - Use Windows Explorer to drag assemblies into the cache.

Assemblies

- An assembly's name is stored in metadata and has a significant impact on the assembly's scope and use by an application
- A strong-named assembly has a fully qualified name that includes the assembly's name, culture, public key, and version number
- The runtime uses this information to locate the assembly and differentiate it from other assemblies with the same name
- A strong-named assembly called myTypes could have the following fully qualified name:

myTypes, Version=1.0.1234.0, Culture="en-US", PublicKeyToken=b77a5c561934e089c

Projects

- In the way it manages files, Visual Studio .NET has only two types of projects - Web and non-Web (local)
- Web project - any project created at an HTTP location in the New Project dialog box
  - These projects are primarily used to provide content to Web browsers (projects known as Web Applications), but are also used when developers want to share data between servers (Web Services)
- Local project -any project created at a non-HTTP location (for example, C:/MyProjects or \\MyComputer\MyProjects).
  - The most common local projects are used to create Windows applications.

Configuration files

- XML files that can be changed as needed
- Developers use configuration files to change settings without recompiling applications
  - Types
    - •Machine.Config- for machine-level settings
    - •Web.Config- for application and directory-level settings.

Machine.Config

- Located in the directory:

  C:\WINDOWS\Microsoft .NET \Framework\version\CONFIG\Machine.config

- Advantage of the Machine.config file
  - Storing settings in the Machine.config file can make your system easier to maintain because you have only one configuration file to search, edit, and maintain

- Disadvantage of the Machine.config file
  - When you deploy a Web application to a new server, the Web application settings that are in the Machine.config file are not copied to the new Web server

Web.Config

- Use the web.config file to share information and settings between Web pages.
- A single web.config file is typically located in the root folder of the Web application.
- You can place additional web.config files in the folder of the virtual directory to which they belong.

Assignment

- Complete student activity 5.2

Microsoft

LESSON 5.3

98-363 Web Development Fundamentals

Publish Web Applications

LESSON 5.3

98-363 Web Development Fundamentals

Lesson Overview

Topics

- Internet Information Server (IIS) installation and configuration
- Web application deployment
- MSI files

Key Terms:

- Internet Information Services – Internet Information Services (IIS) for Windows® Server is a flexible, secure and easy-to-manage Web server for hosting anything on the Web

- MSI File – A Microsoft® Windows® Installer package (.msi) file is a storage file containing the instructions and data required to install an application.

IIS

- In contrast to Active Server Pages (ASP) application settings that are stored in the Microsoft Internet Information Services (IIS) metabase, ASP.NET configuration settings are stored in Extensible Markup Language (XML) files

- This provides advantages when you need to change configuration settings and deploy applications

IIS Advantages

- ASP.NET configuration files are stored in the same directory as the site content
  - At deployment time, copy the content directory to obtain both the content and the application configuration
- Use standard text editors to modify configuration files
  - Easy to change configuration settings both locally and remotely
- The configuration files are extensible; as a result, you can add your own custom configuration settings
- The system automatically detects changes to ASP.NET configuration files
  - No need to restart IIS or reboot the Web server for the settings to take effect.

LESSON 5.3

98-363 Web Development Fundamentals

IIS Configuration

Configure a default Web site

When you install IIS, it is preconfigured to serve as a default Web site; however, you may want to change some of the settings. To change the basic settings for the Web site and to emulate the steps that are required to set up Apache for the first time by using the configuration file:

1. Log on to the Web server computer as an administrator.
2. Click **Start**, point to **Settings**, and then click **Control Panel**.
3. Double-click **Administrative Tools**, and then double-click **Internet Services Manager**.
4. Right-click the Web site that you want to configure in the left pane, and then click **Properties**.
5. Click the **Web site** tab.
6. Type a description for the Web site in the **Description** box.
7. Type the Internet Protocol (IP) address to use for the Web site or leave the **All (Unassigned)** default setting.
8. Modify the Transmission Control Protocol (TCP) port as appropriate.
9. Click the **Home Directory** tab.
10. To use a folder on the local computer, click **A directory on this computer**, and then click **Browse** to locate the folder that you want to use.
11. To use a folder that has been shared from another computer on the network, click **A share located on another computer**, and then either type the network path or click **Browse** to select the shared folder.
12. Click **Read** to grant read access to the folder (required).
13. Click **OK** to accept the Web site properties.

LESSON 5.3

98-363 Web Development Fundamentals

IIS Continued

Create a new Web site

To create a new Web site in Apache, you must set up a virtual host and configure the individual settings for the host. If you are using IIS, you can create a new Web site by translating the following terms to the IIS equivalents:

Apache term	IIS term
DocumentRoot	IIS Web Site Home Directory
ServerName	IIS Host Header
Listen	IIS IP Address and TCP Port

To create a new Web site in IIS, follow these steps:

1. Log on to the Web server computer as an administrator.
2. Click **Start**, point to **Settings**, and then click **Control Panel**.
3. Double-click **Administrative Tools**, and then double-click **Internet Services Manager**.
4. Click **Action**, point to **New**, and then click **Web Site**.
5. After the Web Site Creation Wizard starts, click **Next**.
6. Type a description for the Web site.

   This description is used internally to identify the Web site in Internet Services Manager only.
7. Select the IP address to use for the site.

   If you select **All (unassigned)**, the Web site is accessible on all interfaces and all configured IP addresses.
8. Type the TCP port number to publish the site on.
9. Type the Host Header name (the real name that is used to access this site).
10. Click **Next**.
11. Either type the path to the folder that is holding the Web site documents or click **Browse** to select the folder, and then click **Next**.
12. Select the access permissions for the Web site, and then click **Next**.
13. Click **Finish**.

Assignment

- Complete student activity 5.3

LESSON 5.4

98-363 Web Development Fundamentals

Understand Application Pools

What is an application pool?

- A group of one or more URLs that are served by a worker process or set of worker processes
- Any Web directory or virtual directory can be assigned to an application pool.

How are application pools useful?

- Provide a convenient way to administer a set of Web sites and applications and their corresponding worker processes (because application pools define a set of Web applications that share one or more worker processes)

- Significantly increase both the reliability and manageability of a Web infrastructure

Key Points

- To ensure Web site and Web application isolation, Internet Information Services (IIS) application pools must be configured to have unique names and correctly configured identities

- If errors occur during the configuration of an application pool, the application pool may not be available to serve the Web sites and Web applications that are assigned to it

Key Points

- Web sites and Web applications depend on the availability of Internet Information Services (IIS) application pools
- IIS application pools in turn depend on the Windows Process Activation Service (WAS)
  - If WAS is not running or errors occur during the startup or shutdown of an application pool, Web sites and Web applications may not be available

Key Points

- Internet Information Services (IIS) application pools can be periodically recycled to avoid unstable states that can lead to application crashes, hangs, or memory leaks

- By default, application pool recycling is overlapped, which means that the worker process that is to be shut down is kept running until after a new worker process is started

Assignment

- Complete student activity 5.4

LESSON 5

98-363 Web Development Fundamentals

Complete Quia Test:

MTA WebFund5 Test