

类型安全的系统

类型安全代码

- 防范缓冲区溢出
- 授权方式限制访问内存
- 允许相同进程运行多程序集

应用程序域提供:

- 性能增强
- 代码安全性的增强



代码访问安全

- .NET Framework 安全特性
- 代码访问安全
- 角色访问安全
- 构建安全的ASP.NET Web 应用
- 构建安全的ASP.NET Web Services应用



缓冲区溢出保护

- 应用类型检验防范内存改写
- 在.NET 中System.String的声明和调用没发生改变

```
void CopyString (string src)
{
    stringDest = src;
}
```

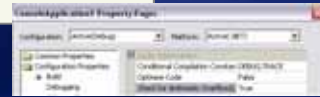
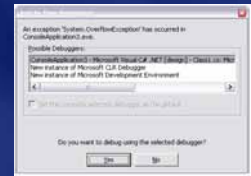
- .NET 中的System.Text.StringBuilder提供了缓冲区边界检查



算法错误检查

- 对算法的错误检查通过:
 - 关键字检查
 - 项目设定

```
byte b=0;
while (true)
{
    Console.WriteLine (b);
    checked
    {
        b++;
    }
}
```



强命名保护

- 强命名
 - 具有唯一的标识符
 - 被用于对程序集的数字签名
- 经过强命名的程序集
 - 防止被篡改
 - 确认程序集原有者的身份
 - 允许组件并行调用

```
sn -k MyFullKey.snk
```



独立存储

- 提供了一个虚拟的文件系统
- 允许设置配额
- 文件的独立存储通过
 - 应用身份确认
 - 用户身份确认

```
IsolatedStorageFile isoStore =
    IsolatedStorageFile.GetUserStoreForAssembly();
```

Internet Zone apps get 10MB by default
Internet 域给应用程序默认分配10MB的空间



基于证据的安全机制

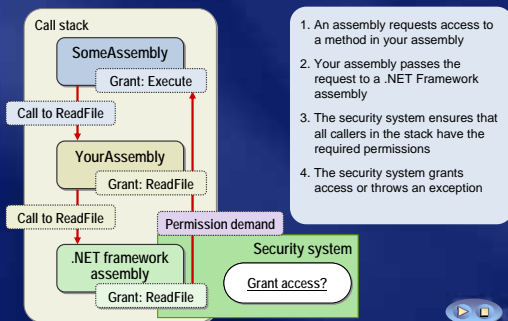
- 证据:
 - 当程序集被访问时评估证据
 - 被用于确定程序集所具有的权限
 - 包含了程序集的:
 - 强命名信息
 - URL (可信, 非可信)
 - Zone (安全区域)
 - 可信的代码签名
 - 用户自定义信息



安全策略

Security entity	Description
Policy	<ul style="list-style-type: none"> • 由管理员进行设定 • 运行时强制检查 • 管理简单 • 包含permissions • 包含code groups
Code group	<ul style="list-style-type: none"> • 一类相似的程序集 • 基于证据 • 与一个permission集合相联系
Permission set	<ul style="list-style-type: none"> • 包含一组经过授权的permission

步入调用堆栈的安全机制



访问许可请求

- 开发者通过许可请求获取必须的权限
- 属性引用实现
- 除非具备了运行程序的最小权限, 否则不能运行

```
//I will only run if I can call unmanaged code
[assembly: SecurityPermission(
    SecurityAction.RequestMinimum,
    UnmanagedCode=true)]
```

Demonstration 1: Code Access Security



- Using the .NET Framework Configuration Tool
- Performing Security Checks
- Requesting Permissions

部分信任的程序集

- .NET Framework 1.1之前的web应用都是full trust 运行
- .NET Framework 1.1提供下面几种信任级别:
 - Full
 - High
 - Medium
 - Low
 - Minimal

角色访问安全

- .NET Framework 安全特性
- 代码访问安全
- 角色访问安全
- 加密
- 构建安全的ASP.NET Web 应用
- 构建安全的ASP.NET Web Services应用

认证和授权

认证需要解决
"Who are you?"
"Am I sure you are who you say you are?"

授权需要解决
"Are you allowed to ... ?"



Identities and Principals

- identity 包含了一个用户的信息，比如用户的登陆名称
- principal 包含了一种角色的信息，比如角色包含的用户和计算机
- .NET Framework 提供了:
 - WindowsIdentity and WindowsPrincipal objects
 - GenericIdentity and GenericPrincipal objects

创建Windows Identities and Principals

- 使用WindowsIdentity和WindowsPrincipal 对象

- Single validation

```
WindowsIdentity myIdent = WindowsIdentity.GetCurrent();
WindowsPrincipal myPrin = new WindowsPrincipal(myIdent);
```

- Repeated validation

```
AppDomain.CurrentDomain.SetPrincipalPolicy(PrincipalPolicy.WindowsPrincipal);
WindowsPrincipal myPrin = (Thread.CurrentPrincipal as WindowsPrincipal);
```

创建 Generic Identities and Principals

- 创建一个 GenericIdentity 和 GenericPrincipal

```
GenericIdentity myIdent = new GenericIdentity("User1");
string[] roles = {"Manager", "Teller"};
GenericPrincipal myPrin =
    new GenericPrincipal(myIdent, roles);
```

- 将创建的GenericPrincipal加载到当前的策略上

```
System.Threading.Thread.CurrentPrincipal = myPrin;
```

执行安全检查

- 在代码中使用Identity and Principal members
 - 比如，使用Identity.Name属性校验用户的登陆名称

```
if (String.Compare(myPrin.Identity.Name, "DOMAIN\Fred",
    true)==0)
{
    // Perform some action
}
```

- 比如，使用类Principal的IsInRole方法检验角色成员

```
if (myPrin.IsInRole("BUILTIN\Administrators"))
{
    // Perform some action
}
```

命令与声明式的安全检查

- Use permissions to perform role-based security checks
 - 命令式

```
PrincipalPermission principal = new PrincipalPermission("Telier", "Manager", true);
try
{
    principal.Demand(); //Does the above match the active principal?
}
```

- 声明式 (通过属性调用实现)

```
[PrincipalPermission(SecurityAction.Demand, Role="Telier", Authenticated=true)]
```

Demonstration 2: Role-Based Security



- Using Windows Role-Based Security
- Using Generic Role-Based Security

构建安全的ASP.NET Web 应用

- .NET Framework 安全特性
- 代码访问安全
- 角色访问安全
- 加密
- 构建安全的ASP.NET Web 应用
- 构建安全的ASP.NET Web Services应用

ASP.NET认证方式

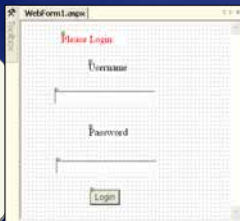
认证方式	优点	缺点
Windows	<ul style="list-style-type: none"> • 使用现有windows的架构 • 控制对于敏感数据的访问权 	<ul style="list-style-type: none"> • 不支持所有的客户端类型 • 每个用户都必须有一个windows账户
Forms	<ul style="list-style-type: none"> • 支持任意客户端 	<ul style="list-style-type: none"> • 需要cookies支持
Microsoft Passport	<ul style="list-style-type: none"> • 提供对于多个Internet Web 站点的一次性登陆验证 • 允许开发人员定制注册页面 	<ul style="list-style-type: none"> • 需要cookies的支持 • 涉及费用问题

基于Forms认证方式的设置

- 设置IIS Anonymous authentication
- 通过设置Web.config 建立Form认证方式
- 建立授权
- 创建一个登陆页面

```
<system.web>
  <authentication mode="Forms">
    <forms>
      loginUrl="WebForm1.aspx"/>
    </authentication>
  </system.web>

  <authorization>
    <deny users="?" />
  </authorization>
</system.web>
```



窗体认证方式的增强

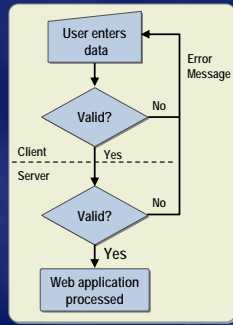
- 开发人员可以通过设置确保cookies安全

```
<authentication mode="Forms">
  <forms loginUrl="login.aspx"
    protection="All"
    requireSSL="true"
    timeout="10"
    name="AppNameCookie"
    path="/FormsAuth"
    slidingExpiration="true"
  </forms>
</authentication>
```

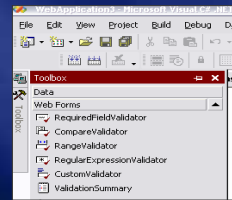
- 开发人员可以创建 application-specific keys来加解密cookies

校验控件

- 客户端校验
 - 提供即时反馈
 - 减少postback
- 服务器校验
 - 重复了客户端的校验
 - 可以结合存储的数据进行验证



校验控件的类型



Demonstration 4: ASP.NET Web Application Security



- Configuring Forms Authentication
- Using Validation Controls

构建安全的ASP.NET Web Services应用

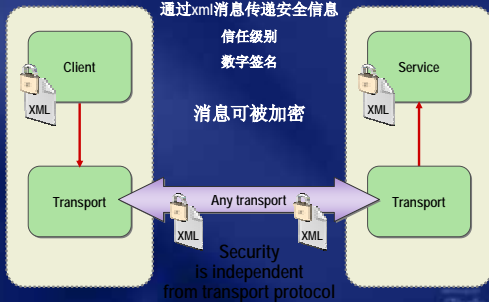
- .NET Framework 安全特性
- 代码访问安全
- 角色访问安全
- 加密
- 构建安全的ASP.NET Web 应用
- 构建安全的ASP.NET Web Services应用

消息级别的安全

通过xml消息传递安全信息


信任级别
数字签名

消息可被加密



Web Service Enhancements (WSE)

- 包含以下内容:
 - SOAP headers包含认证信息
 - 支持消息加密
 - 支持消息数字签名
- 支持消息路由
- 支持附件传递
- 通过程序集 Microsoft.Web.Services.dll 运行



Demonstration 5: Web Services Enhancements

- Implementing Security for a Web Service
- A look at WSE messages

Microsoft TechNet 2003 © 2003

Next Steps

- 1 Stay informed about security
 - Sign up for security bulletins:
http://www.microsoft.com/security/security_bulletins/alerts2.asp
 - Get the latest Microsoft security guidance:
<http://www.microsoft.com/security/guidance/>
- 2 Get additional security training
 - Find online and in-person training seminars:
<http://www.microsoft.com/seminar/events/security.msp>
 - Find a local CTEC for hands-on training:
<http://www.microsoft.com/learning/>

Microsoft TechNet 2003 © 2003

For More Information

- Microsoft Security Site (all audiences)
<http://www.microsoft.com/security>
- MSDN Security Site (developers)
<http://msdn.microsoft.com/security>
- TechNet Security Site (IT professionals)
<http://www.microsoft.com/technet/security>

Microsoft TechNet 2003 © 2003

Questions and Answers

Microsoft TechNet 2003 © 2003

Microsoft®

您的潜力. 我们的动力

Microsoft TechNet 2003 © 2003