

您的潜力. 我们的动力

Microsoft
微软(中国)有限公司

使用ASP.NET AJAX提供的 Authentication Service

赵劼

MSDN特邀讲师

jeffz@live.com

本次课程内容包括

- 使用ASP.NET验证保护Web Service调用
- Authentication Service使用

收听本次课程需具备的条件

- ASP.NET AJAX简单应用

Level 200

ASP.NET Profile

- 可以为每个用户（包括匿名用户）储存信息
- 通过在Web.config中的配置即可在应用程序中使用
- 强类型的属性
- 可以定义属性组
- 可以通过自定义ProfileProvider来扩充

旧事重提: Web Service保护

- 普通情况下, Web Service方法可能会被恶意用户非法调用。
 - 性能
 - 安全性
- 可以使用最常见的ASP.NET验证来保护Web Service
 - Forms Authentication (最常用)

Forms Authentication使用

- ASP.NET 2.0中对于简化了Forms Authentication的使用
 - FormsAuthentication静态类
- 可以使用最常见的ASP.NET验证来保护Web Service
 - Forms Authentication（最常用）

DEMO 1

使用Forms Authentication 保护Web Service调用

Authentication Service

- 提供了AJAX方式进行身份验证（用户登陆）的功能。
- 基于ASP.NET的Membership功能。
 - 可以使用VS 2005提供的ASP.NET 2.0应用程序配置工具来配置

使用Authentication Service

- 出于安全性的考虑, ASP.NET AJAX的 在默认情况下不会开放Authentication Service 功能

```
<authenticationService  
  enabled="true"  
  requireSSL = "true | false" />
```

身份验证功能

- 用户身份验证（登陆）：login方法

```
Sys.Services.AuthenticationService.login(  
    userName, // 用户名  
    password, // 密码  
    isPersistent, // 是否生成持久Cookie  
    customInfo, // 预留字段  
    redirectUrl, // 登陆成功后跳转目标  
    loginCompletedCallback, // 身份验证完成回调函数  
    failedCallback, // 身份验证出错回调函数  
    userContext //用户任意指定的上下文对象  
)
```

身份验证功能 (Cont.)

```
function loginCompletedCallback(  
    validCredentials, //身份验证是否成功  
    userContext, // 用户随意指定的上下文对象  
    methodName // 即“....login”  
) { .. }
```

```
function failedCallback(  
    error, // 错误对象  
    userContext, // 用户随意指定的上下文对象  
    methodName // 即“....login”  
) { .. }
```

注销功能

- 注销功能: `logout`方法

```
System.Services.AuthenticationService.logout(  
    redirectUrl, // 注销后跳转页面  
    logoutCompletedCallback, // 注销成功回调函数  
    failedCallback, // 注销失败回调函数  
    userContext // 用户任意指定的上下文对象  
);
```

注销功能 (Cont.)

```
function logoutCompletedCallback(  
    result, // 预留参数  
    userContext, // 用户随意指定的上下文对象  
    methodName // 即“....logout”  
) { .. }
```

```
function failedCallback(  
    error, // 错误对象  
    userContext, // 用户随意指定的上下文对象  
    methodName // 即“....logout”  
) { .. }
```


您的潜力. 我们的动力

Microsoft
微软(中国)有限公司

DEMO 2

使用Authentication Service

Authentication Service属性




- timeout属性: 设置超时时间
 - `Sys.Services.AuthenticationService.get_timeout();`
 - `Sys.Services.ProfileService.set_timeout(...);`
- defaultLoginCompletedCallback属性
- defaultLogoutCompletedCallback属性
- defaultFailedCallback属性

获取更多MSDN资源


- **MSDN中文网站**
<http://www.microsoft.com/china/msdn>
- **MSDN中文网络广播**
<http://www.msdnwebcast.com.cn>
- **MSDN Flash**
<http://www.microsoft.com/china/newsletter/case/msdn.aspx>
- **MSDN开发中心**
<http://www.microsoft.com/china/msdn/DeveloperCenter/default.mspix>

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。

 **问题和解答 (无问题)**  

在此会议中尚未解答任何问题。

要向演示者提问，请在此处键入问 

提问(A)

删除(D)

问题管理器(Q)

您的潜力. 我们的动力

Microsoft®
微软(中国)有限公司

Microsoft®