

System.Security.Permissions.SecurityPermissionAttribute Class

```
[ILAsm]
.class public sealed serializable SecurityPermissionAttribute extends
System.Security.Permissions.CodeAccessSecurityAttribute

[C#]
public sealed class SecurityPermissionAttribute :
CodeAccessSecurityAttribute
```

Assembly Info:

- *Name:* mscorlib
- *Public Key:* [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00 00]
- *Version:* 2.0.x.x
- *Attributes:*
 - CLSCompliantAttribute(true)

Type Attributes:

- AttributeUsageAttribute(AttributeTargets.Assembly | AttributeTargets.Class | AttributeTargets.Struct | AttributeTargets.Constructor | AttributeTargets.Method, AllowMultiple=true, Inherited=false)

Summary

Used to apply a security action and a set of security permissions to program code.

Inherits From: System.Security.Permissions.CodeAccessSecurityAttribute

Library: BCL

Thread Safety: All public static members of this type are safe for multithreaded operations. No instance members are guaranteed to be thread safe.

Description

[*Note:* The security permissions are defined in the System.Security.Permissions.SecurityPermissionFlag enumeration and are specified using the System.Security.Permissions.SecurityPermissionAttribute.Flags property.

The security information declared by a security attribute is stored in the metadata of the attribute target, and is accessed by the system at run-time. Security attributes are used for declarative security only. For imperative security, use the corresponding permission class, System.Security.Permissions.SecurityPermission.

```
1    The allowable System.Security.Permissions.SecurityPermissionAttribute targets
2    are determined by the System.Security.Permissions.SecurityAction passed to the
3    constructor.
4
5    ]
```

6 **Example**

```
7    In the following example, the attribute target is an assembly. The attribute declares that
8    the ability to assert permissions on behalf of callers is the minimum permission required
9    for the assembly to execute.
10
11    [assembly:SecurityPermissionAttribute(SecurityAction.RequestMinimum,
12    Assertion=true)]
```

13

SecurityPermissionAttribute(System.Security.Permissions.SecurityAction) Constructor

```
[ILAsm]
public rtspecialname specialname instance void .ctor(valuetype
System.Security.Permissions.SecurityAction action)

[C#]
public SecurityPermissionAttribute(SecurityAction action)
```

Summary

Constructs and initializes a new instance of the System.Security.Permissions.SecurityPermissionAttribute class with the specified System.Security.Permissions.SecurityAction value.

Parameters

Parameter	Description
<i>action</i>	A System.Security.Permissions.SecurityAction value.

Exceptions

Exception	Condition
System.ArgumentException	<i>action</i> is not a valid System.Security.Permissions.SecurityAction value.

SecurityPermissionAttribute.CreatePermission() Method

```
[ILAsm]  
.method public hidebysig virtual class System.Security.IPermission  
CreatePermission()  
  
[C#]  
public override IPermission CreatePermission()
```

Summary

Returns a new `System.Security.Permissions.SecurityPermission` object that contains the security information of the current instance.

Return Value

A new `System.Security.Permissions.SecurityPermission` object with the security information of the current instance.

Description

[*Note:* Applications typically do not call this method; it is intended for use by the system.

The security information declared by a security attribute is stored in the metadata of the attribute target, and is accessed by the system at run-time. The system uses the object returned by this method to convert the security information of the current instance into the form stored in metadata.

This method overrides
`System.Security.Permissions.SecurityAttribute.CreatePermission.`

]

SecurityPermissionAttribute.Flags Property

```
[ILAsm]
.property valuetype System.Security.Permissions.SecurityPermissionFlag
Flags { public hidebysig specialname instance valuetype
System.Security.Permissions.SecurityPermissionFlag get_Flags() public
hidebysig specialname instance void set_Flags(valuetype
System.Security.Permissions.SecurityPermissionFlag value) }

[C#]
public SecurityPermissionFlag Flags { get; set; }
```

Summary

Gets or sets values that define the permissions declared by the current instance.

Property Value

One or more System.Security.Permissions.SecurityPermissionFlag values. To specify multiple values in a set operation, use the bitwise OR operator.