System.Net.WebPermission Class

[ILAsm]

```
.class public serializable WebPermission extends
System.Security.CodeAccessPermission
```

[C#]

```
public class WebPermission: CodeAccessPermission
```

Assembly Info:

- *Name:* System
- *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)

Implements:

- **System.Security.IPermission**

Summary

Controls access to Internet resources.

Inherits From: System.Security.CodeAccessPermission

Library: Networking

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

Description

The XML encoding of a `System.Net.WebPermission` instance is defined below in EBNF format. The following conventions are used:

- All non-literals in the grammar below are shown in normal type.

- All literals are in bold font.

The following meta-language symbols are used:

- '*' represents a meta-language symbol suffixing an expression that can appear zero or more times.

- '?' represents a meta-language symbol suffixing an expression that can appear zero or one time.

- '+' represents a meta-language symbol suffixing an expression that can appear one or more times.

- '(',')' is used to group literals, non-literals or a mixture of literals and non-literals.

- '|' denotes an exclusive disjunction between two expressions.

- '::= ' denotes a production rule where a left hand non-literal is replaced by a right hand expression containing literals, non-literals or both.

BuildVersion refers to the build version of the shipping CLI. This is a dotted build number such as '2412.0'.

ECMAPubKeyToken::= `b77a5c561934e089`

HostName refers to a host name such as `www.contoso.com`.

Portnumber denotes a `System.Int32` value indicating a port.

TransportProtocol::= `1` | `2` | `3` /*1= UDP, 2 = TCP, 3 = both */

WebPermissionXML::=

```
<IPermission

class="

System.Net.WebPermission,

System,

Version=1.0.BuildVersion,

Culture=neutral,

PublicKeyToken=ECMAPubKeyToken"

version="1"

(
Unrestricted="true"/>
)
|
>
(<ConnectAccess>
(
<URI>HostName#PortNumber#TransportProtocol</URI>
)+
</ConnectAccess>
</IPermission>
)
|
>
(<AcceptAccess>
(
<URI>HostName#PortNumber#TransportProtocol</URI>
)+


</AcceptAccess>


</IPermission>


)


|


/>
```

WebPermission(System.Security.Permissions.PermissionState) Constructor

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

[C#]
public WebPermission(PermissionState state)
```

Summary

Constructs and initializes a new instance of the `System.Net.WebPermission` class with the specified `System.Security.Permissions.PermissionState` value.

Parameters

Parameter	Description
state	A `System.Security.Permissions.PermissionState` value.

Description

[*Note:* This constructor creates either fully restricted (`System.Security.Permissions.PermissionState.None`) or `System.Security.Permissions.PermissionState.Unrestricted` access to Internet resources.

]

WebPermission(System.Net.NetworkAccess, System.String) Constructor

```
[ILAsm]
public rtspecialname specialname instance void .ctor(valuetype
System.Net.NetworkAccess access, string uriString)

[C#]
public WebPermission(NetworkAccess access, string uriString)
```

Summary

Constructs and initializes a new instance of the `System.Net.WebPermission` class that secures access to the specified URI.

Parameters

Parameter	Description
access	A `System.Net.NetworkAccess` value indicating the type of access to secure.
uriString	A `System.String` that represents the URI to grant access rights to. [*Note:* For more information on the format of this string, see `System.Uri`.]

Exceptions

Exception	Condition
System.ArgumentNullException	*uriString* is `null`.

WebPermission.Copy() Method

[ILAsm]
```
.method public hidebysig virtual class System.Security.IPermission Copy()
```

[C#]
```
public override IPermission Copy()
```

Summary

Returns a new System.Net.WebPermission object containing the same values as the current instance.

Return Value

A new System.Net.WebPermission instance containing the same values as the current instance.

Description

[*Note:* The object returned by this method represents the same access to resources as the current instance.

This method overrides System.Security.CodeAccessPermission.Copy and is implemented to support the System.Security.IPermission interface.

]

WebPermission.FromXml(System.Security.SecurityElement) Method

```
[ILAsm]
.method public hidebysig virtual void FromXml(class
System.Security.SecurityElement securityElement)

[C#]
public override void FromXml(SecurityElement securityElement)
```

Summary

Reconstructs the state of a System.Net.WebPermission object using the specified XML encoding.

Parameters

Parameter	Description
securityElement	A System.Security.SecurityElement instance containing the XML encoding to use to reconstruct the state of a System.Net.WebPermission object.

Description

The state of the current instance is changed to the state encoded in *securityElement*.

[*Note:* For the XML encoding for this class, see the System.Net.WebPermission class page.

This method overrides System.Security.CodeAccessPermission.FromXml.

]

Exceptions

Exception	Condition
System.ArgumentNullException	*securityElement* is null.
System.ArgumentException	*securityElement* does not contain the encoding for a System.Net.WebPermission instance.

WebPermission.Intersect(System.Security.IPermission) Method

```
[ILAsm]
.method public hidebysig virtual class System.Security.IPermission
Intersect(class System.Security.IPermission target)

[C#]
public override IPermission Intersect(IPermission target)
```

Summary

Returns a new `System.Net.WebPermission` object that is the intersection of the current instance and the specified object.

Parameters

Parameter	Description
target	A `System.Net.WebPermission` instance to intersect with the current instance.

Return Value

A new `System.Net.WebPermission` instance that represents the intersection of the current instance and *target*. If the intersection is empty, returns `null`. If the current instance is unrestricted, returns a copy of *target*. If *target* is unrestricted, returns a copy of the current instance.

Description

[*Note:* The intersection of two permissions is a permission that secures the resources and operations secured by both permissions. Specifically, it represents the minimum permission such that any demand that passes both permissions will also pass their intersection.

This method overrides `System.Security.CodeAccessPermission.Intersect` and is implemented to support the `System.Security.IPermission` interface.

]

Exceptions

Exception	Condition
System.ArgumentNullException	*target* is `null`.

System.ArgumentException	*target* is not `null` and is not of type `System.Net.WebPermission`.

WebPermission.IsSubsetOf(System.Security.IPermission) Method

```
[ILAsm]
.method public hidebysig virtual bool IsSubsetOf(class
System.Security.IPermission target)

[C#]
public override bool IsSubsetOf(IPermission target)
```

Summary

Determines whether the current instance is a subset of the specified object.

Parameters

Parameter	Description
target	A System.Net.WebPermission instance that is to be tested for the subset relationship.

Return Value

true if the current instance is a subset of *target*; otherwise, false. If target is null, return false. If the current instance is unrestricted, and *target* is not, returns false. If *target* is unrestricted, returns true.

Description

[*Note:* The current instance is a subset of *target* if the current instance specifies a set of accesses to resources that is wholly contained by *target*. For example, a permission that represents read access to a file is a subset of a permission that represents read and write access to the file.

If this method returns true, the current instance describes access to internet resources that is also described by *target*.

This method overrides System.Security.CodeAccessPermission.IsSubsetOf and is implemented to support the System.Security.IPermission interface.

]

Exceptions

Exception	Condition

System.ArgumentException	*target* is not of type `System.Net.WebPermission`.

WebPermission.ToXml() Method

```
[ILAsm]
.method public hidebysig virtual class System.Security.SecurityElement
ToXml()

[C#]
public override SecurityElement ToXml()
```

Summary

Returns the XML encoding of the current instance.

Return Value

A `System.Security.SecurityElement` containing the XML encoding of the state of the current instance.

Description

[*Note:* For the XML encoding for this class, see the `System.Net.WebPermission` class page.

This method overrides `System.Security.CodeAccessPermission.ToXml`.

]

WebPermission.Union(System.Security.IPermission) Method

```
[ILAsm]
.method public hidebysig virtual class System.Security.IPermission
Union(class System.Security.IPermission target)

[C#]
public override IPermission Union(IPermission target)
```

Summary

Returns a new System.Net.WebPermission object that is the union of the current instance and the specified object.

Parameters

Parameter	Description
target	The System.Net.WebPermission instance to combine with the current instance.

Return Value

A new System.Net.WebPermission instance that represents the union of the current instance and *target*. If the current instance or *target* is unrestricted, returns a System.Net.WebPermission instance that is unrestricted.

Description

[*Note:* The result of a call to System.Net.WebPermission.Union is a permission that represents all of the access to permissions represented by the current instance as well as the permissions represented by *target*. Any demand that passes either the current instance or *target* passes their union.

This method overrides System.Security.CodeAccessPermission.Union and is implemented to support the System.Security.IPermission interface.

]

Exceptions

Exception	Condition
System.ArgumentException	*target* is not null and is not of type System.Net.WebPermission.