

1 System.Net.WebPermission Class

2
3

```
4 [ILASM]  
5 .class public serializable WebPermission extends  
6 System.Security.CodeAccessPermission  
7  
8 [C#]  
9 public class WebPermission: CodeAccessPermission
```

9 Assembly Info:

- 10 • Name: System
- 11 • Public Key: [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00]
- 12 • Version: 1.0.x.x
- 13 • Attributes:
 - 14 ○ CLSCompliantAttribute(true)

15 Implements:

- 16 • System.Security.IPermission

17 Summary

18

19 Controls access to Internet resources.

20 Inherits From: System.Security.CodeAccessPermission

21

22 Library: Networking

23

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

26

27 Description

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

- 30 • All non-literals in the grammar below are shown in normal type.
- 31 • All literals are in bold font.

32 The following meta-language symbols are used:

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

- 1 • '?' represents a meta-language symbol suffixing an expression
2 that can appear zero or one time.
- 3 • '+' represents a meta-language symbol suffixing an expression
4 that can appear one or more times.
- 5 • '(',')' is used to group literals, non-literals or a mixture of
6 literals and non-literals.
- 7 • '|' denotes an exclusive disjunction between two expressions.
- 8 • '::=' denotes a production rule where a left hand non-literal is
9 replaced by a right hand expression containing literals, non-
10 literals or both.

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

13
14 ECMAPubKeyToken ::= **b77a5c561934e089**

15
16 HostName refers to a host name such as `www.contoso.com`.

17
18 Portnumber denotes a **System.Int32** value indicating a port.

19
20 TransportProtocol ::= **1 | 2 | 3** /*1 = UDP, 2 = TCP, 3 = both */

21
22 WebPermissionXML ::=

23
24
25 **<IPermission**

26
27
28 **class="**

29
30
31 **System.Net.WebPermission,**

32
33
34 **System,**

35
36
37 **Version=1.0.BuildVersion,**

38
39
40 **Culture=neutral,**

41
42
43 **PublicKeyToken=ECMAPubKeyToken"**

44
45
46 **version="1"**

47
48

```
1      (
2
3
4      Unrestricted="true"/>
5
6
7      )
8
9
10     |
11
12
13     >
14
15
16     ( <ConnectAccess>
17
18     (
19
20
21
22     <URI>HostName#PortNumber#TransportProtocol</URI>
23
24
25     )+
26
27
28     </ConnectAccess>
29
30
31     </IPermission>
32
33
34     )
35
36
37     |
38
39
40     >
41
42
43     ( <AcceptAccess>
44
45     (
46
47
48
49     <URI>HostName#PortNumber#TransportProtocol</URI>
50
51
52     )+
53
54
55     </AcceptAccess>
56
```

```
1
2     </IPermission>
3
4
5     )
6
7
8     |
9
10
11    />
12
13
```

1 WebPermission(System.Security.Permissions.PermissionState) Constructor

```
3 [ILASM]  
4 public rtspecialname specialname instance void  
5 .ctor(valuetype System.Security.Permissions.PermissionState  
6 state)  
  
7 [C#]  
8 public WebPermission(PermissionState state)
```

9 Summary

10 Constructs and initializes a new instance of the
11 **System.Net.WebPermission** class with the specified
12 **System.Security.Permissions.PermissionState** value.

13 Parameters

14
15

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

16

17 Description

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

22

1 WebPermission(System.Net.NetworkAcce 2 ss, System.String) Constructor

```
3 [ILASM]  
4 public rtsspecialname specialname instance void  
5 .ctor(valuetype System.Net.NetworkAccess access, string  
6 uriString)  
  
7 [C#]  
8 public WebPermission(NetworkAccess access, string  
9 uriString)
```

10 Summary

11 Constructs and initializes a new instance of the
12 **System.Net.WebPermission** class that secures access to the
13 specified URI.

14 Parameters

Parameter	Description
<i>access</i>	A System.Net.NetworkAccess value indicating the type of access to secure.
<i>uriString</i>	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 .]

17 18 Exceptions

Exception	Condition
System.ArgumentNullException	<i>uriString</i> is null .

21
22
23

1 WebPermission.Copy() Method

```
2 [ILASM]  
3 .method public hidebysig virtual class  
4 System.Security.IPermission Copy()  
  
5 [C#]  
6 public override IPermission Copy()
```

7 Summary

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

10 Return Value

11

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

14 Description

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

17 This method overrides
18 **System.Security.CodeAccessPermission.Copy** and is implemented
19 to support the **System.Security.IPermission** interface.]
20

21

1 WebPermission.FromXml(System.Security 2 .SecurityElement) Method

```
3 [ILASM]  
4 .method public hidebysig virtual void FromXml(class  
5 System.Security.SecurityElement securityElement)  
  
6 [C#]  
7 public override void FromXml(SecurityElement  
8 securityElement)
```

9 Summary

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

12 Parameters

13
14

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

15
16

16 Description

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

19
20
21
22
23
24

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

This method overrides **System.Security.CodeAccessPermission.FromXml.**]

25 Exceptions

26
27

Exception	Condition
System.ArgumentNullException	<i>securityElement</i> is null .
System.ArgumentException	<i>securityElement</i> does not contain the encoding for a System.Net.WebPermission instance.

28
29
30

1 WebPermission.Intersect(System.Security 2 .IPermission) Method

```
3 [ILASM]  
4 .method public hidebysig virtual class  
5 System.Security.IPermission Intersect(class  
6 System.Security.IPermission target)  
7  
8 [C#]  
9 public override IPermission Intersect(IPermission target)
```

9 Summary

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

12 Parameters

13
14

Parameter	Description
<i>target</i>	A System.Net.WebPermission instance to intersect with the current instance.

15
16
17

16 Return Value

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

23 Description

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

28
29 This method overrides
30 **System.Security.CodeAccessPermission.Intersect** and is
31 implemented to support the **System.Security.IPermission**
32 interface.]

33 Exceptions

34
35

1
2
3

Exception	Condition
<code>System.ArgumentNullException</code>	<i>target</i> is null .
<code>System.ArgumentException</code>	<i>target</i> is not null and is not of type System.Net.WebPermission .

1 WebPermission.IsSubsetOf(System.Security.IPermission) Method

```
3 [ILASM]  
4 .method public hidebysig virtual bool IsSubsetOf(class  
5 System.Security.IPermission target)  
  
6 [C#]  
7 public override bool IsSubsetOf(IPermission target)
```

8 Summary

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

11 Parameters

12
13

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

14
15
16

15 Return Value

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

20 Description

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

26
27 If this method returns **true**, the current instance describes access to
28 internet resources that is also described by *target*.

29
30 This method overrides
31 **System.Security.CodeAccessPermission.IsSubsetOf** and is
32 implemented to support the **System.Security.IPermission**
33 interface.]

34 Exceptions

35
36

1
2
3

Exception	Condition
System.ArgumentException	<i>target</i> is not of type System.Net.WebPermission .

1 WebPermission.ToXml() Method

```
2 [ILASM]  
3 .method public hidebysig virtual class  
4 System.Security.SecurityElement ToXml()  
  
5 [C#]  
6 public override SecurityElement ToXml()
```

7 Summary

8 Returns the XML encoding of the current instance.

9 Return Value

10

11 A **System.Security.SecurityElement** containing the XML encoding of
12 the state of the current instance.

13 Description

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

16
17 This method overrides
18 **System.Security.CodeAccessPermission.ToXml.**]

19

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
<i>target</i>	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.ArgumentNullException	<i>target</i> is null .
System.ArgumentException	<i>target</i> is not null and is not of type System.Net.WebPermission .

1
2