# System.Net.SocketPermissionAttribute Class

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

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

**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)

**Type Attributes:**

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

**Summary**

Used to declaratively specify security actions to control socket connections.

**Inherits From: System.Security.Permissions.CodeAccessSecurityAttribute**

**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 properties of a `System.Net.SocketPermissionAttribute` are required to have non-null values. Once set, the values of the properties cannot be changed.

[*Note:* The details of a socket connection are specified using the properties of the current instance. For example, to secure a socket connection to port 80, set the `System.Net.SocketPermissionAttribute.Port` property equal to "80".

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.Net.SocketPermission`.

The allowable `System.Net.SocketPermissionAttribute` targets are determined by the `System.Security.Permissions.SecurityAction` passed to the constructor.

]

# SocketPermissionAttribute(System.Security.Permissions.SecurityAction) Constructor

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


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

## Summary

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

## Parameters

| Parameter | Description |
|-----------|-------------|
| *action* | A `System.Security.Permissions.SecurityAction` value. |

## Permissions

| Permission | Description |
|------------|-------------|
| **System.ArgumentException** | *action* is not a valid `System.Security.Permissions.SecurityAction` value. |

# SocketPermissionAttribute.CreatePermission() Method

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

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

**Summary**

Returns a `System.Net.SocketPermission` that contains the security information of the current instance.

**Return Value**

A `System.Net.SocketPermission` object with the security information of the current instance.

**Description**

[*Note:* This method overrides `System.Security.Permissions.SecurityAttribute.CreatePermission`.

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

The security information described 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.

]

**Exceptions**

| Exception | Condition |
|---|---|
| **System.ArgumentException** | One or more of the current instance's `System.Net.SocketPermissionAttribute.Access`, `System.Net.SocketPermissionAttribute.Host`, `System.Net.SocketPermissionAttribute.Transport` or `System.Net.SocketPermissionAttribute.Port` properties is `null`. |

# SocketPermissionAttribute.Access Property

```
[ILAsm]
.property string Access { public hidebysig specialname instance
string get_Access() public hidebysig specialname instance void
set_Access(string value) }


[C#]
public string Access { get; set; }
```

**Summary**

Gets or sets the network access method specified by the current instance.

**Property Value**

A `System.String` containing a network access method allowed by the current instance. Valid values are "Accept" and "Connect".

**Description**

This property is write-once. Once this property has been set to a non-null value, attempts to set this property to new value cause a `System.ArgumentException`.

Valid values for this property correspond to `System.Net.NetworkAccess` enumeration values.

**Exceptions**

| Exception | Condition |
|---|---|
| **System.ArgumentException** | `System.Net.SocketPermissionAttribute.Access` is being set and is not `null`. |

# SocketPermissionAttribute.Host Property

```
[ILAsm]
.property string Host { public hidebysig specialname instance string
get_Host() public hidebysig specialname instance void
set_Host(string value) }

[C#]
public string Host { get; set; }
```

## Summary

Gets or sets the DNS host name or IP address specified by the current instance.

## Property Value

A `System.String` containing a DNS host name or IP address.

## Description

This property is write-once. Once this property has been set to a non-null value, attempts to set this property to new value cause a `System.ArgumentException`.

## Exceptions

| Exception | Condition |
|---|---|
| **System.ArgumentException** | `System.Net.SocketPermissionAttribute.Host` is being set and is not `null`. |

# SocketPermissionAttribute.Port Property

```
[ILAsm]
.property string Port { public hidebysig specialname instance string
get_Port() public hidebysig specialname instance void
set_Port(string value) }

[C#]
public string Port { get; set; }
```

**Summary**

Gets or sets the port specified by the current instance.

**Property Value**

A `System.String` containing a port number, or "All" or -1 to indicate all ports.

**Description**

This property is write-once. Once this property has been set to a non-null value, attempts to set this property to new value cause a `System.ArgumentException`.

**Exceptions**

| Exception | Condition |
|---|---|
| **System.ArgumentException** | `System.Net.SocketPermissionAttribute.Port` is being set and is not `null`. |

# SocketPermissionAttribute.Transport Property

```
[ILAsm]
.property string Transport { public hidebysig specialname instance
string get_Transport() public hidebysig specialname instance void
set_Transport(string value) }


[C#]
public string Transport { get; set; }
```

## Summary

Gets or sets the transport type specified by the current instance.

## Property Value

A `System.String` containing the transport type associated with the current instance. Valid values are "All", "Connectionless", "ConnectionOriented", "Tcp", and "Udp".

## Description

This property is write-once. Once this property has been set to a non-null value, attempts to set this property to new value cause a `System.ArgumentException`.

[*Note:* Valid values for this property correspond to `System.Net.TransportType` enumeration values.]

## Exceptions

| Exception | Condition |
|---|---|
| **System.ArgumentException** | `System.Net.SocketPermissionAttribute.Transport` is being set and is not `null`. |