

System.Net.IAuthenticationModule Interface

```
[ILAsm]
.class interface public abstract IAuthenticationModule

[C#]
public interface IAuthenticationModule
```

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)

Summary

Implemented by types that perform client authentication.

Library: Networking

Description

The `System.Net.IAuthenticationModule` interface defines the properties and methods that types are required to implement to handle client authentication. Types that implement this interface are called authentication modules. In addition to implementing this interface, an authentication module implements an authentication protocol, such as the Kerberos protocol. The `System.Net.IAuthenticationModule.AuthenticationType` property value is a case-insensitive string that typically indicates the protocol implemented by the module. Each authentication module registered with the authentication manager is required to have a unique `System.Net.IAuthenticationModule.AuthenticationType`. The following string values are reserved for use by modules implementing the indicated protocols:

AuthenticationType	Protocol
"basic"	Basic as defined by IETF RFC 2617
"digest"	Digest access as defined by IETF RFC 2617
"kerberos"	Kerberos as defined by IETF RFC 1510

[*Note:* Authentication modules are registered with the authentication manager (`System.Net.AuthenticationManager`) by calling the `System.Net.AuthenticationManager.Register` method. When the authentication manager receives an authentication request, registered authentication modules are

1 given the opportunity to handle the authentication in their
2 `System.Net.IAuthenticationModule.Authenticate` method. Similarly, when a client
3 wishes to avoid waiting for the server to request authentication, it can request
4 preauthentication information to send with a request. If the
5 `System.Net.IAuthenticationModule.CanPreAuthenticate` property of a registered
6 module returns `true`, it is among the modules that are given the opportunity to provide
7 the preauthentication information via the
8 `System.Net.IAuthenticationModule.PreAuthenticate` method.

9
10 Not all modules receive all authentication and preauthentication requests. The
11 authentication manager searches for an authentication module by invoking the
12 `System.Net.IAuthenticationModule.Authenticate` or
13 `System.Net.IAuthenticationModule.PreAuthenticate` method of each registered
14 module in the order in which it was registered. Once a module returns a
15 `System.Net.Authorization` instance, indicating that it handles the authentication, the
16 authentication manager terminates the search.

17
18]

19

IAuthenticationModule.Authenticate(System.String, System.Net.WebRequest, System.Net.ICredentials) Method

```
[ILAsm]  
.method public hidebysig virtual abstract class System.Net.Authorization  
Authenticate(string challenge, class System.Net.WebRequest request, class  
System.Net.ICredentials credentials)
```

```
[C#]  
Authorization Authenticate(string challenge, WebRequest request,  
ICredentials credentials)
```

Summary

Returns an instance of the `System.Net.Authorization` class that provides a response to an authentication challenge.

Parameters

Parameter	Description
<i>challenge</i>	A <code>System.String</code> containing the authentication challenge sent by the server. The content of this string is determined by the authentication protocol(s) used by the server that issued the challenge.
<i>request</i>	The <code>System.Net.WebRequest</code> instance that received <i>challenge</i> .
<i>credentials</i>	The credentials of the <code>System.Net.WebRequest</code> instance that received <i>challenge</i> .

Return Value

A `System.Net.Authorization` instance containing the challenge response, or `null` if the challenge cannot be handled.

Behaviors

If the authentication module can handle *challenge*, this method proceeds with the authentication in accordance with the authentication protocol implemented by the current instance and returns a `System.Net.Authorization` instance containing the challenge response. If the authentication module cannot handle the challenge, this method returns `null`. If the authentication module encounters an error while conducting the authentication process, this method can, but is not required to throw an exception.

1

2 **How and When to Override**

3 Implement this method to receive and optionally handle requests for client
4 authentication from the authentication manager.

5

6 **Usage**

7 The `System.Net.AuthenticationManager` class invokes the
8 `System.Net.IAuthenticationModule.Authenticate` method on registered
9 authentication modules to allow modules to handle a server challenge.

10

11

1
2 **IAuthenticationModule.PreAuthenticate(System.Net.WebRequest, System.Net.ICredentials)**
3 **Method**
4

```
5 [ILAsm]  
6 .method public hidebysig virtual abstract class System.Net.Authorization  
7 PreAuthenticate(class System.Net.WebRequest request, class  
8 System.Net.ICredentials credentials)  
  
9 [C#]  
10 Authorization PreAuthenticate(WebRequest request, ICredentials  
11 credentials)
```

12 **Summary**

13 Returns an instance of the System.Net.Authorization class containing client
14 authentication information.

15 **Parameters**

Parameter	Description
<i>request</i>	The System.Net.WebRequest instance associated with the authentication request.
<i>credentials</i>	The credentials associated with the authentication request.

16
17 **Return Value**

18 A System.Net.Authorization instance containing client authentication information to
19 be sent with *request*, or null if the current instance does not support
20 preauthentication.

21 **Behaviors**

22 This method returns an instance of the System.Net.Authorization class containing
23 authentication information to be sent with the request. This method is required to return
24 null if it cannot handle preauthentication requests from the authentication manager.

25
26 **How and When to Override**

27 Implement this method if the protocol implemented by the current instance supports
28 preauthentication.

1

2 **Usage**

3 This method is used by the authentication manager to handle requests by clients for
4 authentication information that will be used to preempt a server's request for
5 authentication. This method is not called unless the
6 `System.Net.IAuthenticationModule.CanPreAuthenticate` property returns true.

7

8

9

10

1 IAuthenticationModule.AuthenticationType 2 Property

```
3 [ILAsm]  
4 .property string AuthenticationType { public hidebysig virtual abstract  
5 specialname string get_AuthenticationType() }  
  
6 [C#]  
7 string AuthenticationType { get; }
```

8 Summary

9 Gets the authentication type of the current instance.

10 Property Value

11 A System.String indicating the authentication type of the current authentication
12 module.

13 Description

14 This property is read-only.

15 Behaviors

16 The following string values are reserved for use by modules implementing the indicated
17 protocols:

AuthenticationType	Protocol
"basic"	Basic as defined by IETF RFC 2617
"digest"	Digest access as defined by IETF RFC 2617
"kerberos"	Kerberos as defined by IETF RFC 1510

18
19

20 How and When to Override

21 Implement this property to return a string that identifies all instances of the current
22 type. The string returned by this property typically indicates the protocol implemented
23 by the current type. For example, an authentication module that implements the Digest
24 protocol would typically return "digest" as its AuthenticationType.

25 Usage

1 The `System.Net.IAuthenticationModule.AuthenticationType` property must be
2 unique for all registered authentication modules. The value of this property is used by
3 the `System.Net.AuthenticationManager.Register` method to determine if there is
4 already an authentication module registered for the type. Modules can also be
5 unregistered by passing the value of this property to the
6 `System.Net.AuthenticationManager.Unregister` method.

7

8

1 IAuthenticationModule.CanPreAuthenticate 2 Property

```
3 [ILAsm]  
4 .property bool CanPreAuthenticate { public hidebysig virtual abstract  
5 specialname bool get_CanPreAuthenticate() }  
  
6 [C#]  
7 bool CanPreAuthenticate { get; }
```

8 Summary

9 Gets a `System.Boolean` value indicating whether the authentication module supports
10 preauthentication.

11 Property Value

12 `true` if the authentication module supports preauthentication; otherwise `false`.

13 Description

14 This property is read-only.

15 Behaviors

16 The `System.Net.IAuthenticationModule.CanPreAuthenticate` property is set to `true`
17 to indicate that the authentication module can respond with a
18 `System.Net.Authorization` instance when the
19 `System.Net.IAuthenticationModule.PreAuthenticate` method is called. This property
20 returns `false` if the protocol implemented by the current authentication module does
21 not support preauthentication.

22

23 Usage

24 This method is used by the authentication manager to determine which modules receive
25 requests by clients for authentication information that will be supplied in anticipation of
26 a server's request for authentication. This method controls whether the
27 `System.Net.IAuthenticationModule.PreAuthenticate` method can be called.

28

29