

System.Net.AuthenticationManager Class

```
[ILASM]
.class public AuthenticationManager extends System.Object

[C#]
public class AuthenticationManager
```

Assembly Info:

- *Name:* System
- *Public Key:* [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00]
- *Version:* 1.0.x.x
- *Attributes:*
 - CLSCompliantAttribute(true)

Summary

Manages the authentication modules called during the client authentication process.

Inherits From: System.Object

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 **System.Net.AuthenticationManager** class manages authentication modules that are responsible for client authentication. [Note: The **System.Net.AuthenticationManager** is called by **System.Net.WebRequest** instances to provide information that is sent to servers to authenticate the client. The authentication process may consist of requests to an authentication server separate from the resource server, as well as any other activities required to properly authenticate a client.]

The **System.Net.AuthenticationManager** queries registered authentication modules by calling the **System.Net.IAuthenticationModule.Authenticate** method for each module. The first authentication module that returns a **System.Net.Authorization** instance is used to authenticate the request. An authentication module, which can be any object that implements the **System.Net.IAuthenticationModule** interface, is registered using the **System.Net.AuthenticationManager.Register** method. Authentication modules are called in the order in which they

- 1 are registered.
- 2
- 3 Applications typically do not access this type directly; it provides
- 4 authentication services for the **System.Net.WebRequest** type.
- 5

1 AuthenticationManager.Authenticate(System.String, System.Net.WebRequest, System.Net.ICredentials) Method

```
4 [ILASM]  
5 .method public hidebysig static class  
6 System.Net.Authorization Authenticate(string challenge,  
7 class System.Net.WebRequest request, class  
8 System.Net.ICredentials credentials)  
  
9 [C#]  
10 public static Authorization Authenticate(string challenge,  
11 WebRequest request, ICredentials credentials)
```

12 Summary

13 Calls registered authentication modules to find a module that responds
14 to the specified authentication challenge.

15 Parameters

Parameter	Description
<i>challenge</i>	A System.String containing the challenge returned by an Internet host. The content of this string is determined by the authentication protocol(s) used by the server that issued the challenge.
<i>request</i>	The System.Net.WebRequest that received <i>challenge</i> .
<i>credentials</i>	The System.Net.ICredentials associated with <i>request</i> . [Note: The System.Net.WebRequest.Credentials property of <i>request</i> is used to supply this argument.]

19 Return Value

21 A **System.Net.Authorization** instance containing the response from
22 the authentication module, or **null** if no authentication module
23 responded to the challenge.

24 Applications do not call this method; it is called by
26 **System.Net.WebRequest** instances.

27 Description

28 The **System.Net.AuthenticationManager.Authenticate** method
29 invokes the **System.Net.IAuthenticationModule.Authenticate**
30 method of each registered authentication module until one of the
31 modules returns a **System.Net.Authorization** instance.
32 Authentication modules are called in the order in which they were

1 registered via the **System.Net.AuthenticationManager.Register**
2 method.

3
4 [Note: The **System.Net.Authorization.Message** property of the
5 object returned by this method contains the client's response to the
6 server challenge contained in *challenge*.]

7 **Exceptions**

8
9

Exception	Condition
System.ArgumentNullException	<i>challenge, request, or credentials</i> is null .

10
11
12

AuthenticationManager.PreAuthenticate(System.Net.WebRequest, System.Net.ICredentials) Method

```
[ILASM]
.method public hidebysig static class
System.Net.Authorization PreAuthenticate(class
System.Net.WebRequest request, class
System.Net.ICredentials credentials)

[C#]
public static Authorization PreAuthenticate(WebRequest
request, ICredentials credentials)
```

Summary

Attempts to obtain a **System.Net.Authorization** instance used to initiate client authentication.

Parameters

Parameter	Description
<i>request</i>	A System.Net.WebRequest containing a request for access to a resource.
<i>credentials</i>	The System.Net.ICredentials associated with <i>request</i> .

Return Value

A **System.Net.Authorization** instance if an authentication module can provide authentication information to be sent with *request*; otherwise, **null**. If *credentials* is **null**, this method returns **null**.

Description

The **System.Net.AuthenticationManager.PreAuthenticate** method invokes the **System.Net.IAuthenticationModule.PreAuthenticate** method of each registered authentication module until one of the modules returns a **System.Net.Authorization** instance. Authentication modules are called in the order in which they were registered via the **System.Net.AuthenticationManager.Register** method.

[Note: The **System.Net.Authorization** instance contains the information that will be sent by a client to initiate authentication instead of waiting for the server to request it. Authentication modules

1 that support preauthentication allow clients to improve server
2 efficiency by avoiding extra round trips caused by authentication
3 challenges.]

4
5 [Note: If an authorization module supports preauthentication of
6 requests, its
7 **System.Net.IAuthenticationModule.CanPreAuthenticate** property
8 returns **true**. Note that preauthentication requires that an
9 authentication of the client has already occurred. The information
10 obtained from the initial authentication is used to provide the
11 **System.Net.Authorization.Message** that is sent to the server as an
12 authentication header in *request*.]

13 **Exceptions**

14
15

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

16
17
18

1 AuthenticationManager.Register(System. 2 Net.IAuthenticationModule) Method

```
3 [ILASM]  
4 .method public hidebysig static void Register(class  
5 System.Net.IAuthenticationModule authenticationModule)  
  
6 [C#]  
7 public static void Register(IAuthenticationModule  
8 authenticationModule)
```

9 Summary

10 Adds an authentication module to the list of registered authentication
11 modules managed by the authentication manager.

12 Parameters

13
14

Parameter	Description
<i>authenticationModule</i>	The System.Net.IAuthenticationModule to register.

15
16

16 Description

17 The **System.Net.AuthenticationManager.Register** method adds an
18 authentication module to the end of the list of modules managed by
19 the authentication manager. Each registered module is required to
20 have a unique
21 **System.Net.IAuthenticationModule.AuthenticationType**. If a
22 module with the same
23 **System.Net.IAuthenticationModule.AuthenticationType** is
24 already registered, this method removes the registered module, and
25 adds *authenticationModule* to the end of the list. [*Note:* Authentication
26 modules are called in the order in which they were added to the list.]
27
28 [*Note:* To remove a module, call one of the
29 **System.Net.AuthenticationManager.Unregister** methods.]

30 Exceptions

31
32

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

33
34
35

1 AuthenticationManager.Unregister(System.Net.IAuthenticationModule) Method

```
3 [ILASM]
4 .method public hidebysig static void Unregister(class
5 System.Net.IAuthenticationModule authenticationModule)
6
7 [C#]
8 public static void Unregister(IAuthenticationModule
authenticationModule)
```

9 Summary

10 Removes the specified authentication module from the list of
11 registered modules.

12 Parameters

13
14

Parameter	Description
<i>authenticationModule</i>	The System.Net.IAuthenticationModule module to remove.

15
16

16 Description

17 [Note: To add an authentication module to the list of registered
18 modules, call the **System.Net.AuthenticationManager.Register**
19 method.]

20 Exceptions

21
22

Exception	Condition
System.ArgumentNullException	<i>authenticationModule</i> is null .
System.InvalidOperationException	<i>authenticationModule</i> is not a registered authentication module.

23
24
25

1 AuthenticationManager.Unregister(System.String) Method

```
3 [ILASM]  
4 .method public hidebysig static void Unregister(string  
5 authenticationScheme)  
  
6 [C#]  
7 public static void Unregister(string authenticationScheme)
```

8 Summary

9 Removes the authentication module with the specified authentication
10 type from the list of registered modules.

11 Parameters

12
13

Parameter	Description
<i>authenticationScheme</i>	A System.String containing the authentication type of the module to remove.

14
15

15 Description

16 The *authenticationScheme* is required to match the value returned by
17 the **System.Net.IAuthenticationModule.AuthenticationType**
18 property of a registered authentication module.

19
20
21
22

[Note: To add an authentication module to the list of registered modules, call the **System.Net.AuthenticationManager.Register** method.]

23 Exceptions

24
25

Exception	Condition
System.ArgumentNullException	<i>authenticationScheme</i> is null .
System.InvalidOperationException	There is no registered module with the <i>authenticationScheme</i> authentication type.

26
27
28

1 AuthenticationManager.RegisteredModule 2 s Property

```
3 [ILASM]  
4 .property class System.Collections.IEnumerator  
5 RegisteredModules { public hidebysig static specialname  
6 class System.Collections.IEnumerator  
7 get_RegisteredModules() }  
  
8 [C#]  
9 public static IEnumerator RegisteredModules { get; }
```

10 Summary

11 Gets a list of registered authentication modules.

12 Property Value

13

14 A **System.Collections.IEnumerator** that provides access to the list
15 of registered authentication modules.

16 Description

17 This property is read-only.

18

19 [Note: The **System.Net.AuthenticationManager.Register** method
20 adds modules to the list of registered authentication modules, and the
21 **System.Net.AuthenticationManager.Unregister** method removes
22 modules from it.]

23