Sign In
Join
Search
IIS
Home
Downloads
Learn
Reference
Solutions
Technologies
.NET Framework
ASP.NET
PHP
Media
Windows Server
SQL Server
Web App Gallery
Windows Azure
lools
Visual Studio
Expression Studio
Windows Internet Explorer
Cot Holp
Ack a Question in our Forums
Ask a Question in our Forums
Rloge
Forume
i oi unis

Home > Blogs > Microsoft Support Team's IIS Blog > Configuring Many-to-One Client Certificate Mappings for IIS 7/7.5

Configuring Many-to-One Client Certificate Mappings for IIS 7/7.5	Chause their De st	Microsoft Support Team's IIS Blog
Posted: Apr 27, 2010 2 comments	Share this Post	IIS Search My Blog
Average Rating		Web Contact Mo
Tags Authentication Client Certificate IIS7 IIS7.5	Email this Post	Topics Contact Me
	DotNetKicks	Recent Posts
	Digg	MSDeploy - Changing the
	Facebook	ConnectionString
	Del.icio.us	Event ID:1074 and Event ID:1310
		Visual Studio 2010 – My Favorite

Many-to-one Client certificate mapping is used by the Internet Information Services (IIS) to associate an end user to a windows account when the client certificate is used for the user authentication. The user session is executed under the context of this mapped windows account by IIS. For this to work we need to ensure that the certificate to account mapping is configured correctly in IIS.

In IIS 6.0, the user had the option to configure Many-to-One client certificate mapping through the IIS Manager User Interface. In IIS 7/7.5, we don't have such an interface for either One-to-One or Many-to-One mappings. This post talks about the Configuration Editor IIS 7/7.5 extension that can be used to achieve the mappings either for One-to-One or Many-to-One. Here we will talk in specific about Many-to-1 mapping.

## IIS 7 or IIS 7.5 Schema

This is the schema for the IIS Client Certificate Mapping authentication feature in IIS 7 or IIS 7.5.

<pre>sectionSchema name="system.webServer/security/authentication/iisClientCertificateMappingAuthentication"&gt;</pre>
<pre><attribute defaultvalue="true" name="many/ounecertificatemappingschabied" type="bool"></attribute></pre>
<pre><element name="manyToOneMappings"></element></pre>
<collection addelement="add" clearelement="clear"></collection>
<attribute isuniquekey="true" name="name" required="true" type="string" validationtype="nonEmptyString"></attribute>
<attribute name="description" type="string"></attribute>
<attribute defaultvalue="true" name="enabled" type="bool"></attribute>
<attribute defaultvalue="Allow" name="permissionMode" type="enum"></attribute>
<pre><enum name="Allow" value="1"></enum></pre>
<pre><enum name="Deny" value="2"></enum></pre>
<pre><element name="rules"></element></pre>
<collection addelement="add" clearelement="clear"></collection>
<attribute iscombinedkey="true" name="certificateField" required="true" type="enum"></attribute>
<pre><enum name="Subject" value="1"></enum> ```````````````````````````````````</pre>
<pre><enum name="Issuer" value="2"></enum></pre>
<pre><attribute casesensitive="true" iscombinedkey="true" name="certificateSubField" required="true" type="string"></attribute> <attribute casesensitive="true" iscombinedkey="true" name="matchCriteria" required="true" type="string"></attribute></pre>
<attribute defaultvalue="true" iscombinedkey="true" name="compareCaseSensitive" type="bool"></attribute>
<pre><attribute name="userName" type="string" validationtype="nonEmptyString"></attribute></pre>
<pre><attribute casesensitive="true" defaultvalue="[enc:AesProvider::enc]" encrypted="true" name="password" type="string"></attribute></pre>



</collection> </element> ... </sectionSchema>

These are the prerequisites needed for this walkthrough.

- 1. We have installed IIS Client Certificate Mapping module on the server.
- 2. A Web Site is configured with an HTTPS binding which can accept SSL connections.
- 3. We have a client certificate installed on the client.
- 4. IIS 7 Administration Pack is installed on the IIS 7.0 server. NOTE: Configuration Editor is shipped by default on IIS 7.5.

### Walkthrough

Step 1:

1. Launch the IIS manager and select your web site which is being configured for client certificate authentication.

2. In the features View select Configuration Editor under Management section in the Features View.



3. Go to "system.webServer/security/authentication/iisClientCertificateMappingAuthentication" in the drop down box as shown below:



You will see a window to configure Many-to-One or One-to-One certificate mappings here. This is the UI provided through Configuration editor from where we can setup all the mapping configurations.

# ٢

## Configuration Editor

Section: ity/authentication/iisClientCertificateMappingA	Authentication  • From: ApplicationHost.config <location path="Default Web Site"></location> •
Deepest Path: MACHINE/WEBROOT/APPHOST	
defaultLogonDomain	
enabled	False
logonMethod	ClearText
manyToOneCertificateMappingsEnabled	True
manyToOneMappings	(Count=0)
oneToOneCertificateMappingsEnabled	True
oneToOneMappings	(Count=0)

4. We can go ahead and modify the properties through this GUI.

- Set **enabled** to true
- Set manyToOneCertificateMappingsEnabled to True
- Select manyToOneMappings and click on the extreme end at the Ellipsis button to launch the new window for configuring mappings.

5. Under this new window go ahead and Add a new item. You can modify the properties from within the window as shown below:

description	1st User Mapping	
enabled	True	
name	My 1st Mapping	
password	•••••	
permissionMode	Allow	
rules	(Count=0)	
userName	mydomain\testuser	

6. Click on the Ellipsis button for rules and this will give you an option to add multiple patterns for matching based on certificate properties.

Properties:	
certificateField	Subject
certificateSubField	* CN
compareCaseSensitive	🔋 True
matchCriteria	* Test User

P	roperties:		
	certificateField	ş	Issuer
	certificateSubField	P	CN
	compareCaseSensitive	ę	True
	matchCriteria	P	My IT Enterprise

Iter	ems:				
	certificateField	certificateSubField	matchCriteria	compareCaseSensitive	Entry Path
	Subject	CN	Test User	True	MACHINE/WEBROOT/APPHOST
	Issuer	CN	My IT Enterprise	True	MACHINE/WEBROOT/APPHOST

So here above we have two entries for rules for mapping the certificate. In the above case we are using two different fields named Subject and the Issuer in the certificate field and based on the **matchcriteria** property map the certificate to the account mydomain\testuser.

Shown below is how the final mapping for a specific windows account looks like. As you can see there are two entries for rules for this account.

8 23 Collection Editor - system.webServer/security/authentication/iisClientCertificateMappingAuthentication/manyToOneMappings/ Items: Actions: enabled permissionMode userName name description password Entry Path Collection My 1st Mapping 1st User Mapping True Allow mydomain\testuser LS1setup! MACHINE/WEBROOT/APPHOST Add Clear All **Item Properties** • Lock Item X Remove **Properties:** Help 1st User Mapping description Online Help enabled True \* My 1st Mapping name password ..... permissionMode Allow rules (Count=2) userName mydomain\testuser rules

Similarly we can have other mappings for various accounts based on the fields "Issuer" and "Subject" in the Certificate.

We can also use Configuration Editor to configure One-to-One mapping. One can follow the instructions in the article listed below to configure One-to-One mappings.

#### http://learn.iis.net/page.aspx/478/configuring-one-to-one-client-certificate-mappings/

## Appendix

So far what we have seen is achieved using the Configuration Editor which gives you a graphical interface to easily set the configuration.

You can achieve the same thing using APPCMD command, in fact the Configuration Editor does the same thing in the background and adds these settings in the ApplicationHost.config file. Configuration Editor gives you an option to run these commands manually, it generates the scripts to achieve that from its UI itself.



These are the Code snippets to perform the same steps as above to configure mapping. They were was generated using Configuration Editor's Script Generation feature.

#### AppCmd instructions

```
appcmd.exe set config "Default Web Site" -section:system.webServer/security/authentication/iisClientCertificateMappingAuthentication / en
appcmd.exe set config "Default Web Site" -section:system.webServer/security/authentication/iisClientCertificateMappingAuthentication /+"
appcmd.exe set config "Default Web Site" -section:system.webServer/security/authentication/iisClientCertificateMappingAuthentication /+"
1
C# Code
using System;
using System.Text;
using Microsoft.Web.Administration;
internal static class Sample -
      private static void Main() {
            using(ServerManager serverManager = new ServerManager()) {
                 Configuration config = serverManager.GetApplicationHostConfiguration();
                  ConfigurationSection iisClientCertificateMappingAuthenticationSection = config.GetSection("system.webServer/security/authenticat:
                  iisClientCertificateMappingAuthenticationSection["enabled"] = true
                 iisClientCertificateMappingAuthenticationSection["manyToOneCertificateMappingsEnabled"] = true;
                 ConfigurationElementCollection manyToOneMappingsCollection = iisClientCertificateMappingAuthenticationSection.GetCollection("many
                  ConfigurationElement addElement = manyToOneMappingsCollection.CreateElement("add");
                 addElement["name"] = @"My 1st Mapping";
addElement["description"] = @"1st User Mapping";
addElement["userName"] = @"mydomain\testuser";
addElement["password"] = @"abcdef";
                  ConfigurationElementCollection rulesCollection = addElement.GetCollection("rules");
                  ConfigurationElement addElement1 = rulesCollection.CreateElement("add");
                  addElement1["certificateField"] = @"Subject";
addElement1["certificateSubField"] = @"CN";
addElement1["matchCriteria"] = @"Test User";
                  rulesCollection.Add(addElement1);
                  manyToOneMappingsCollection.Add(addElement);
                  serverManager.CommitChanges();
            }
      }
```

Scripting (JavaScript)	
<pre>var adminNanger = new ActiveXObject('Microoft.ApplicationHost.WritableAdminNanger'); daminManger.CommitPath = "WACHTEKVERBOOT(APPHOST; var isClientCertificateMappingAuthenticationSection Properties. Item('manyToOneCertificateMappingsEnabled').Value = true; isClientCertificateMappingAuthenticationSection.Properties.Item('manyToOneCertificateMappingsEnabled').Value = true; var anyToOneMappingSCollection = from the Mapping'; addElement.Properties.Item('name').Value = "Ny It Mapping'; addElement.Properties.Item('custName').Value = "abcdef'; var ulesCollection = ulesCollection.CreateMeedTement('add'); addElement.Properties.Item('custName').Value = "abcdef'; var ulesCollection = ulesCollection.CreateMeedTement('add'); addElement.Properties.Item('custName').Value = "bogetf'; addElement.Properties.Item('custName').Value = "Subject'; addElement1 = rulesCollection.AddElement() addElement1.Properties.Item('custName').Value = "Test User'; rulesCollection.AddElement(addElement); addElement1.AddElement(addElement); addElement1.AddElement(addElement); addElement1.AddElement(addElement); addElement1.AddElement(addElement); addElement1.AddElement(addElement); addElement1.AddElement(addElement); addElement1.AddElement(addElement); addElement1.AddElement(addElement); addElement1.AddElement(addElement); addElement1.AddElement(addElement); addElement1.AddElement(addElement); addElement1.AddElement(addElement); addElement2.AddElement(addElement); addElement2.AddElement(addElement); addElement2.AddElement(addElement); addElement2.AddElementElement(addElement); addElement2.AddElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElementElemElementElementElementElementElementElementEle</pre>	
Read the complete post here	
Submit a Comment	

You are logged in as Anonymous

}

Plain text is accepted. URLs starting with **http://** are converted to links.

Submit Comment



This site is hosted for Microsoft by Neudesic, LLC. | © 2012 Microsoft. All rights reserved. Privacy Statement Terms of Use Contact Us Advertise With Us Follow us on: Twitter Facebook Feedback on IIS Powered by IIS8 Microsoft

http://blogs.iis.net/webtopics/archive/2010/04/27/configuring-many-to-one-client-certificat... 2/13/2013