

## Re: WSE 3.0 + UserNameToken without X.509 Cert/Kerberos + Signing + Encryption How?

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.webservices.enhancements/2006-03/>

---

- *From:* "Pablo Cibraro" <[pcibraro@xxxxxxxxxxxxx](mailto:pcibraro@xxxxxxxxxxxxx)>
  - *Date:* Wed, 22 Mar 2006 10:26:08 -0300
- 

Hi James,

First of all, the UsernameOverTransport turn-key assertion does not use message security and therefore it does not encrypt the message.

You need to combine this assertion with a secure transport like SSL if you want to protect your messages.

WSE does not provide a turn-key assertion to encrypt messages using a username token only, so you will have to develop a custom assertion in order to do that.

Secondly, you should add an authorization assertion to restrict the access to your webmethods.

Thirdly, secure conversation does not optimize the amount of data sent over the wire. This article in my weblog gives a description of this feature

<http://weblogs.asp.net/cibrax/archive/2006/02/21/438670.aspx>

All the turn-key assertions that use message security add the headers required by WS-Security to the message, and it is difficult to optimize that part. You should use an assertion for transport security instead.

This security guide from the Pattern & Practices team also provides really good information about this topic

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnpag2/html/WSSP.asp>

I hope this can help you.

Regards,

Pablo Cibraro

<http://weblogs.asp.net/cibrax>

"James Hancock" <[~jamie@xxxxxxxxxxxxxxxxxxxxx](mailto:~jamie@xxxxxxxxxxxxxxxxxxxxx)> wrote in message [news:%23jGmjmwTGHA.736@xxxxxxxxxxxxxxxxxxxxx](mailto:news:%23jGmjmwTGHA.736@xxxxxxxxxxxxxxxxxxxxx)

Re: WSE 3.0 + UserNameToken without X.509 Cert/Kerberos + Signing + Encryption How?

OK, thanks to Julia's help and a bunch of stuff in the newsgroups I've got some of this working but not all...

Basically I need to be able to encrypt both sides of the communication between client and server using a UserNameToken that passes the UserName but not the password but is a shared secret. I cannot use X.509 or anything like that because the number of client computers is unknown and could easily be thousands, and the server may not necessarily have an SSL key and I don't want to leave it up to my clients to optionally get an SSL key, because most of them won't because they're lazy or cheap or some other "really good reason" that will jeopardize their data. <aside>Hence Shared secret, which I used in WSE 2.0 with great effect because the shared secret password was unique per client computer, and it was a 512 bit hash which is really hard to break, and after every operation the hash would change (sent in the encrypted message body by the server to the client and then stored in a different encryption in the local database). (Yes, I know, if you break one message, you break 'em all, but you can't use two messages to compare between the two... I was working on an algorithm that the server and client would use to calculate the new hash without one sending it to the other, but since this works, and is very secure the way it is, it wasn't high on the list of priorities...</aside>

I was able to do this stuff in WSE 2.0 before using the now depreciated Context.Security... stuff. No go in WSE 3.0 because it's depreciated. If I use the depreciated functions, it works fine... I think... can't tell because it doesn't use the policy stuff of course.

I've got the signing stuff done and working by using a Custom UserNameTokenManager on the server and then creating the right entries in my web.config file. Which works and is called and validates the whole deal.. assuming the client request adds a proper UserNameToken...

It still lets clients that don't use anything query my WebMethods however.

So I looked in the newsgroups here and decided I needed a policy file because the way I was doing it before was to query the Context.Security stuff looking for the entries... which is depreciated.

So I added the following on the server:

```
<policies xmlns="http://schemas.microsoft.com/wse/2005/06/policy>  
  
<extensions>
```

Re: WSE 3.0 + UserNameToken without X.509 Cert/Kerberos + Signing + Encryption How?

```
<extension name="usernameOverTransportSecurity"
type="Microsoft.Web.Services3.Design.UsernameOverTransportAssertion,
Microsoft.Web.Services3, Version=3.0.0.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" />

<extension name="wse3Trace" type="WSETracingFilter.WSEFilterAssertion,
WSETracingFilter, Version=3.0.0.0, Culture=neutral,
PublicKeyToken=aa253a6b9020c4eb"/>

</extensions>

<policy name="RegistrationPolicy">

<usernameOverTransportSecurity />

<requireActionHeader />

<wse3Trace/>

</policy>

</policies>
```

Then on the Service itself:

```
[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
```

```
[Policy("RegistrationPolicy")]
```

```
public class Registration : System.Web.Services.WebService {
```

```
...
```

and the following on the client:

```
<policies xmlns="http://schemas.microsoft.com/wse/2005/06/policy">

<extensions>

<extension name="usernameOverTransportSecurity"
type="Microsoft.Web.Services3.Design.UsernameOverTransportAssertion,
Microsoft.Web.Services3, Version=3.0.0.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" />

<extension name="wse3Trace" type="WSETracingFilter.WSEFilterAssertion,
WSETracingFilter, Version=3.0.0.0, Culture=neutral,
PublicKeyToken=aa253a6b9020c4eb"/>
```

Re: WSE 3.0 + UserNameToken without X.509 Cert/Kerberos + Signing + Encryption How?

</extensions>

<policy name="UserNamePolicy">

<usernameOverTransportSecurity />

</policy>

</policies>

and in the client request:

Microsoft.Web.Services3.Security.Tokens.UsernameToken tok = new  
Microsoft.Web.Services3.Security.Tokens.UsernameToken(Username, Password,  
Microsoft.Web.Services3.Security.Tokens.PasswordOption.SendNone);

r.SetPolicy("UserNamePolicy");

r.SetClientCredential(tok);

Note that I'm using Mike's tool to monitor the requests.... hence the  
added  
Extension...I tried removing it and users can still just request the  
WebMethod and it works.

Sorry if this is a dupe everyone. It didn't seem to go through the first  
time...

All is good in that Mike's tool tells me that the username token is added  
etc. However!

1. The contents of the message are not encrypted. They absolutely have to  
be  
encrypted.... But the docs say that UsernameoverTransport doesn't support  
it, but when I query the UserNameToken it says that it does support  
encryption... Help please!

2. You can still request the WebMethods without any of this stuff and it  
works.

Also...

I would like to make sure that the minimum amount of data is sent over the  
wire (there could be dialup users using this!) so any suggestions on how  
to  
get this whole system to send stuff only once on the first request etc.  
would be greatly appreciated. (i.e. how do I enable SecureConversation???)  
Is  
that even the right way to go about it?)

I wish MS would just accept that this is a scenario that they need to

Re: WSE 3.0 + UserNameToken without X.509 Cert/Kerberos + Signing + Encryption How?

support and put it in instead of making it harder and harder every time they do this, cause people are just going to get around it (like me) and thus likely are not going to get it right and think they have encrypted secure communications when they don't... better to have MS do it and do it right instead of letting people hack something together...

Thanks for any help you might be able to provide!

James Hancock