

# Re: Commercial Certificate

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.webservices.enhancements/2005-09/>

---

- *From:* "William Stacey [MVP]" <[staceyw@xxxxxxxx](mailto:staceyw@xxxxxxxx)>
  - *Date:* Thu, 8 Sep 2005 17:54:48 -0400
- 

Thanks Cormac. FWIW, I created a new one based on Secure Remote Password (SRP) protocol from Tom Wu of Stanford:  
<http://channel9.msdn.com/ShowPost.aspx?PostID=107763>

This basically does the same thing as the other one, as the end result is still a SCT; but it does the key exchange very different.

I feel this one is even more secure than the other one for various reasons:

- 1) No RSA public key to protect or worry about on the client side. Keys are dynamically generated for each run of the protocol.
- 2) No keys on \*either side need to be stored or generated.
- 3) Discovering the password does not allow previous (or future) messages to be decrypted by an attacker.
- 4) Discovery of the SKey, does not get anyone closer to discovering the password.
- 5) No password is ever sent on the wire, not even encrypted. Only Proof of Knowledge of the password. Password is never in the server pipe-line.
- 6) No replay on the SCT negotiation is possible. So it is kinda like a built in Nonce, without having to use or track Nonce history.
- 7) Works perfect with custom username/pw database.

Possible down sides:

- 1) Sends Username in the clear.
- 2) As Password is never sent in the clear (nor can it be generated) so the server side can not use LogonUser() API to do Windows logon in your logon provider.

However, both of those could be overcome by using another step. Get a SCT using an Anonymous Username and some well-known password. Then use that SCT to encrypt another token such as UsernameToken.SendPlainText or some other method. As you have a SCT, you can encrypt a message and reply to do pretty much what ever you wanted. Hope that makes some sense. Cheers.

--

William Stacey [MVP]

--

William Stacey [MVP]

Re: Commercial Certificate

"Cormac" <Cormac@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message [news:69598075-C5F5-42E2-BD8B-3AC169E322B8@xxxxxxxxxxxxxxxxxxxx](mailto:news:69598075-C5F5-42E2-BD8B-3AC169E322B8@xxxxxxxxxxxxxxxxxxxx)

> Hi just me again  
>  
> Excellent point Julie, have seen some of your posts, blog and I think some  
> articles you may have written. I still think William's implementation is  
> the  
> best solution I have seen anywhere because certificates are messy to set  
> up  
> on client machines  
> where in Williams scenario strong named assemblies are used I have found  
> they are far more simple to enable. Plus the whole point of Secure  
> Conversation is that it is far easier to implement and not restricted like  
> SSL  
> in its capacity for just endpoint to endpoint encryption. I.E. what if  
> more  
> hops are required in the solution.  
>  
> I have been playing with the new WSE 3.0, the implementation of MTOM and  
> the  
> turnkey scenario's are excellent but very disappointed that still no  
> choice  
> for someone who DOES NOT WANT TO USE X509 certificates, Kerberos, or SSL  
> and  
> just plain jane (sorry only jane I knew wasn't very plain) solution like  
> what  
> William has come up with that is super powerful as well as being simple.  
>  
> Cheers  
>  
> Cormac  
>  
> "Julie Lerman" wrote:  
>  
>> wow – some storm. The power just came back on! <G>  
>>  
>> Anyway...  
>>  
>> depending on your scenario, you don't always need the clients to have  
>> their  
>> own certificates, though you definitely want one on the server. Typical  
>> scenario is if the clients' are being authenticated either on the  
>> intranet  
>> using their windows logins or over the web with a login/password against  
>> a  
>> database. You can use something like a secure conversation and get all of  
>> the encryption and signing. there are situations where this might mean  
>> encrypting and signing with a username token which is possible but not  
>> recommended – and if you are talking about WSE2.0 (assuming this to be  
>> the

Re: Commercial Certificate

Re: Commercial Certificate

>> case) and considering using the username tokens – definitely check Keith  
>> Brown's article about using them safely. (you should find that right on  
>> the  
>> msdn web services (Securing the Username Token with Web Services  
>> Enhancements 2.0 ) at msdn.microsoft.com/webservices/building/wse.  
>>  
>> You really have to figure out what it is you want and need to accomplish  
>> in  
>> your application (on both ends) and then you can decide how you want to  
>> put  
>> the pieces together. It is a little complicated which is why in WSE3.0,  
>> they  
>> have gone to a model of selecting the entire scenario from one end to the  
>> other and back again, rather than determining what you want the client to  
>> do  
>> and then separately determining what you want the server to do. That's  
>> the  
>> new turnkey security scenarios.  
>>  
>> Anyway – I hope this helps a little, and if you want to explain what your  
>> scenario is, I can try to help you figure out where you need what types  
>> of  
>> certificates. Also, if you are able to move right to WSE3.0 (which means  
>> using VS2005 and also not deploying until late fall) then a lot of these  
>> things will be much easier.  
>>  
>> Julie Lerman  
>>  
>> "Julie Lerman" <jlermanATNOSPAMPLEASethedatafarm.com> wrote in message  
>> [news:uskaHHuIFHA.3380@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:uskaHHuIFHA.3380@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)  
>> > Guys–  
>> > Do you NEED certificates on all of the clients?  
>> > The most common scenario is to get a web server certificate. This  
>> > confused  
>> > me at first because there is "no such thing" at verisign/thawte etc.  
>> > They  
>> > are SSL Certificates!!  
>> >  
>> > oops there's lightning!!!  
>> > gotta shut down  
>> > more later  
>> > julie lerman  
>> > "Alex Trebek" <trebek@xxxxxxxxxxxxxxxx> wrote in message  
>> > [news:b22b9\\$42e690f2\\$d844140d\\$3594@xxxxxxxxxxx](mailto:news:b22b9$42e690f2$d844140d$3594@xxxxxxxxxxx)  
>> >> Excellent!! -- Thanks!!!  
>> >>  
>> >> Alex  
>> >>  
>> >>  
>> >> "Cormac" <Cormac@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message  
>> >> [news:510627EC-8DE6-4662-8204-FEFAF5D20539@xxxxxxxxxxxxxxxxxxxx](mailto:news:510627EC-8DE6-4662-8204-FEFAF5D20539@xxxxxxxxxxxxxxxxxxxx)

Re: Commercial Certificate

>> >>> Hi Alex/Sam  
>> >>>  
>> >>> I was in a similar situation since I didn't want to even use X509  
>> >>> certificates tried to find a resolution to using X509 certificates  
>> >>> since  
>> >>> you  
>> >>> have to install them on all client machines, if you get them from a  
>> >>> certificate authority they cost a packet. If you create your own then  
>> >>> you  
>> >>> have to create your own certificate authority and issue them through  
>> >>> one  
>> >>> of  
>> >>> the Microsoft servers (forgot the name). Until I found William  
>> >>> Staceys  
>> >>> (Cool  
>> >>> Guy) blog.  
>> >>>  
>> >>> <http://spaces.msn.com/members/staceyw/Blog/cns!1pnsZpX0fPvDxLKC6rAAhLsQ!268.entry>  
>> >>>  
>> >>> He was mad enough to come up with a solution that uses Security  
>> >>> Context  
>> >>> Tokens or Secure Conversation as many people call it that allows the  
>> >>> developer to develop a Security Context Token Service that issues  
>> >>> Security  
>> >>> Context Tokens to clients and encrypt and sign each SOAP message  
>> >>> without  
>> >>> using X509 certificates.  
>> >>>  
>> >>> He uses strong naming on each assembly to create a Public and  
>> >>> Private  
>> >>> key  
>> >>> just like in X509 certificates to create a Symmetric key to be used  
>> >>> by  
>> >>> both  
>> >>> endpoints.  
>> >>>  
>> >>> I have implemented it with WSE 2.0 SP 3 and am upgrading it to Beta  
>> >>> 2, I  
>> >>> would strongly recommend it instead of using X509 certificates why  
>> >>> through  
>> >>> money and a lot of frustration away on X509 certificates when this is  
>> >>> free  
>> >>> and  
>> >>> better in my humble opinion.  
>> >>>  
>> >>> Cormac  
>> >>>  
>> >>> "Alex Trebek" wrote:  
>> >>>  
>> >>>> If anyone has some insight here, I'd appreciate it as well.. Versign  
>> >>>> was not

Re: Commercial Certificate

